



# Étalonnage des robots à câbles : identification et qualification

Julien Alexandre Dit Sandretto

## ► To cite this version:

Julien Alexandre Dit Sandretto. Étalonnage des robots à câbles : identification et qualification. Autre [cs.OH]. Université Nice Sophia Antipolis, 2013. Français. <NNT : 2013NICE4059>. <tel-00933816>

**HAL Id: tel-00933816**

**<https://tel.archives-ouvertes.fr/tel-00933816>**

Submitted on 21 Jan 2014

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

UNIVERSITÉ DE NICE - SOPHIA ANTIPOLIS  
ÉCOLE DOCTORALE STIC  
SCIENCES ET TECHNOLOGIES DE L'INFORMATION  
ET DE LA COMMUNICATION

# T H È S E

pour obtenir le titre de

**Docteur en Sciences**

de l'Université de Nice - Sophia Antipolis

**Mention : INFORMATIQUE**

Présentée et soutenue par

Julien ALEXANDRE DIT SANDRETTO

## Étalonnage des robots à câbles identification et qualification

Thèse dirigée par David DANEY et Gilles TROMBETTONI

préparée à l'INRIA Sophia Antipolis, Projet COPRIN

soutenue le 10 septembre 2013

### **Jury :**

*Rapporteurs :* GANGLOFF, JACQUES - Professeur à l'Université de Strasbourg  
JAULIN, LUC - Professeur à l'ENSTA Bretagne  
*Directeur :* TROMBETTONI, GILLES - Professeur à l'Université de Montpellier  
*Co-directeur :* DANEY, DAVID - Chargé de recherche à l'INRIA  
*Président :* RUEHER, MICHEL - Professeur à l'université de Nice  
*Examineur :* GOUTTEFARDE, MARC - Chargé de recherche CNRS au LIRMM





This work is licensed under the Creative Commons Attribution - Pas d'Utilisation Commerciale - Pas de Modification 3.0 France License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-nc-nd/3.0/fr/>.





## Remerciements

La thèse est terminée et le moment le plus agréable est arrivé, la rédaction de mes remerciements. Je vais profiter de ces quelques lignes pour exprimer ma sincère reconnaissance à l'ensemble des personnes qui m'ont accompagné de près ou d'un peu plus loin tout au long de ces trois années de thèse.

En premier lieu, je tiens à adresser mes remerciements à Gilles Trombettoni, directeur de cette thèse et à David Daney, mon encadrant, pour leurs conseils précieux et la liberté qu'ils m'ont laissée tout au long de mes travaux. J'ai beaucoup appris à leur contact, aussi bien sur leur domaine de travaux respectif que sur le monde de la recherche. Je les remercie également de m'avoir donné ma chance malgré mon parcours atypique.

Ensuite, je remercie très chaleureusement Jean-Pierre Merlet pour m'avoir accueilli au sein de son équipe et pour son immense savoir qu'il est toujours disponible à partager malgré un emploi du temps surchargé. J'ai également beaucoup apprécié le côtoyer dans des moments moins formels où il ne parvient plus à cacher son immense gentillesse.

Je souhaite remercier Jacques Gangloff et Luc Jaulin d'avoir accepté la lourde charge d'être rapporteurs. En juillet et en août ils auraient pu préférer la plage à la lecture de ce manuscrit.

Je remercie Michel Rueher de m'avoir fait l'honneur de présider le jury de soutenance, ainsi que Marc Gouttefarde d'être venu en tant qu'examinateur. Marc a également joué un rôle important dans cette thèse en dirigeant le projet ANR Co-GiRo et en permettant la construction du robot de grande échelle étudié dans ce manuscrit, malgré les nombreuses difficultés inhérentes à un projet d'une telle envergure.

Cette thèse s'est déroulée au sein d'une équipe formidable, je tiens donc à témoigner ma gratitude à Odile, Yves et Bertrand, avec qui j'ai parfois discuté de science mais surtout avec qui j'ai pu respirer dans les moments difficiles en parlant musique, cinéma ou bonne chère. J'adresse également des remerciements particuliers à Nathalie, toujours disponible et amicale, qui n'a malheureusement pas pu m'accompagner durant les trois années de ma thèse. Je remercie bien sûr les doctorants de l'équipe, avec qui on a pu aussi bien s'entraider face à un problème que se détendre face à la mer.

Et pour finir, je remercie mes parents qui m'ont toujours soutenu ainsi que les amis (ils se reconnaîtront) que je n'ai pas pu voir autant que je le souhaitais, mais qui ont su me faire comprendre qu'ils étaient toujours là.

Voici une citation de circonstances :

**“L’incertitude n’est pas dans les choses mais dans notre tête : l’incertitude est une méconnaissance.”**

de *Jacques Bernouilli*, Extrait du *Ars conjectandi*



# Table des matières

<b>Introduction générale</b>	<b>1</b>
<b>Notations et conventions</b>	<b>3</b>
<b>1 Définitions et revue de littérature</b>	<b>5</b>
1.1 Les manipulateurs . . . . .	5
1.1.1 Les manipulateurs parallèles . . . . .	6
1.1.2 Les robots parallèles à câbles . . . . .	8
1.2 Amélioration de la précision . . . . .	18
1.2.1 Généralités sur l'étalonnage . . . . .	20
1.2.2 Méthodes pour l'identification de paramètres . . . . .	24
1.2.3 Dans le cadre de l'étalonnage . . . . .	28
1.3 Analyse par intervalles . . . . .	28
1.3.1 Arithmétique par intervalles . . . . .	29
1.3.2 Schéma de Newton par intervalles . . . . .	30
1.3.3 Systèmes linéaires d'intervalles . . . . .	30
1.3.4 Caractérisation de la solution . . . . .	32
1.3.5 Méthodes de traitement des systèmes d'intervalles . . . . .	32
1.3.6 Contracteurs . . . . .	35
1.4 Conclusion . . . . .	40
<b>2 Résolution des systèmes non linéaires sur-contraints d'intervalles, étalonnage d'un mécanisme simple</b>	<b>41</b>
2.1 Schéma de Newton sur-contraint . . . . .	42
2.2 Caractérisation de la solution du système linéarisé sur-contraint . . . . .	42
2.2.1 Approche des moindres carrés . . . . .	43
2.2.2 Considération du reste . . . . .	44
2.2.3 Caractérisation de la solution . . . . .	45
2.3 Algorithmes et nouvelle définition du résultat de l'étalonnage . . . . .	47
2.3.1 Calcul d'une boîte extérieure . . . . .	47
2.3.2 Calcul d'une boîte intérieure . . . . .	47
2.3.3 Résultat de l'étalonnage . . . . .	48
2.4 Étalonnage d'un mécanisme simple . . . . .	48
2.4.1 Présentation du mécanisme . . . . .	48
2.4.2 Identification . . . . .	48
2.4.3 Mesures, incertitudes et estimée initiale . . . . .	49
2.5 Approches classiques . . . . .	51
2.5.1 Barycentre des sous systèmes carrés . . . . .	51
2.5.2 Système complet et la jacobienne . . . . .	52
2.5.3 Moindres carrés . . . . .	52
2.6 Approche des intervalles . . . . .	52
2.6.1 Données et leur incertitude . . . . .	52
2.6.2 Identification des paramètres avec les intervalles . . . . .	52
2.7 Discussion et conclusion . . . . .	56

<b>3</b>	<b>Modélisation des robots à câbles</b>	<b>61</b>
3.1	Modèle d'un câble . . . . .	62
3.1.1	Câble indéformable de masse nulle . . . . .	63
3.1.2	Câble élastique de masse nulle . . . . .	63
3.1.3	Câble élastique de masse non nulle . . . . .	64
3.2	Modèle géométrico-statique . . . . .	65
3.2.1	Modèle géométrique implicite . . . . .	65
3.2.2	Transmission des efforts pour les modèles de câble non pesant . . . . .	65
3.2.3	Transmission des efforts pour le modèle de câble pesant . . . . .	67
3.2.4	Modèle géométrique complet et réaliste . . . . .	67
3.2.5	Choix du modèle pour l'étalonnage . . . . .	68
3.3	Simplification du modèle de câble . . . . .	69
3.3.1	Vérification de l'hypothèse . . . . .	69
3.3.2	Incertitude du modèle de câble . . . . .	73
3.3.3	Espace de travail . . . . .	74
3.4	Conclusion . . . . .	77
<b>4</b>	<b>Étalonnage des robots parallèles à câbles</b>	<b>79</b>
4.1	Modélisation . . . . .	80
4.1.1	Étalonnage . . . . .	81
4.1.2	Auto-étalonnage . . . . .	82
4.2	Identifiabilité . . . . .	84
4.2.1	Identifiabilité dans la littérature . . . . .	84
4.2.2	Quelques définitions . . . . .	85
4.2.3	Recherche de poses optimales . . . . .	86
4.2.4	Identifiabilité après la prise de mesure . . . . .	87
4.3	Expérimentations . . . . .	89
4.3.1	Mesures . . . . .	89
4.3.2	Procédure d'étalonnage du prototype ReelAx8 . . . . .	91
4.3.3	Procédure d'étalonnage du robot CoGiRo . . . . .	95
4.4	Discussion . . . . .	100
<b>5</b>	<b>Intervalles : étalonnage certifié du robot CoGiRo</b>	<b>103</b>
5.1	Modélisation du robot CoGiRo . . . . .	104
5.2	Identifiabilité . . . . .	106
5.2.1	Nombre de solutions . . . . .	106
5.2.2	Méthode . . . . .	106
5.2.3	Résultats expérimentaux . . . . .	107
5.3	Identification avec le système linéarisé sur-contraint d'intervalles . . . . .	108
5.3.1	Méthode . . . . .	108
5.3.2	Résultats expérimentaux . . . . .	108
5.3.3	Validation de l'étalonnage . . . . .	109
5.3.4	Discussion . . . . .	111
5.4	Identification avec une approche programmation par contraintes . . . . .	112
5.4.1	Différentes approches pour considérer les incertitudes de modélisation du câble . . . . .	112
5.4.2	Contraction locale des paramètres en fonction d'une pose (le cas contrainte par contrainte) . . . . .	114

5.4.3	Contraction locale des paramètres en fonction d'un sous système formé de trois poses (le cas bien contraint) . . . . .	114
5.4.4	Identification avec la <b>q-intersection</b> . . . . .	116
5.4.5	Résultats expérimentaux . . . . .	118
5.4.6	Validation de l'étalonnage . . . . .	123
5.4.7	Discussion . . . . .	123
5.5	Conclusion . . . . .	125
<b>Conclusion générale</b>		<b>127</b>
<b>Bibliographie</b>		<b>131</b>
<b>A Procédure d'étalonnage</b>		<b>145</b>
A.1	Introduction . . . . .	145
A.2	Préparation . . . . .	145
A.3	Acquisition des mesures . . . . .	146
A.3.1	Diagnostic comportemental . . . . .	146
A.3.2	Acquisition . . . . .	146
A.4	Précautions particulières . . . . .	146
<b>B Caractéristiques du prototype ReelAx8</b>		<b>147</b>
<b>C Caractéristiques du robot CoGiRo</b>		<b>149</b>
<b>D Listes des poses pour l'étalonnage des robots</b>		<b>151</b>
D.1	Poses pour le robot ReelAx8 . . . . .	151
D.2	Poses pour le robot CoGiRo . . . . .	152
<b>Index</b>		<b>155</b>



# Table des figures

1.1	Architecture sérielle (a) et parallèle (b) [Khalil 1999]	6
1.2	La plate-forme de Gough (Dunlop Tyre)	7
1.3	Un simulateur de vol basé sur l'architecture de la plate-forme de Gough-Stewart (Lufthansa)	7
1.4	Un robot Delta industriel : le FlexPicker (ABB Flexible Automation)	8
1.5	Une grue médiévale reconstruite	9
1.6	Une marionnette birmane du XV <sup>e</sup> siècle	10
1.7	Vitrail de l'église du Sacré-Cœur de Menton	10
1.8	Manipulateur à 8 câbles et 6 ddl en configuration suspendue	11
1.9	Manipulateur à 8 câbles et 6 ddl en configuration pleinement contrainte	12
1.10	ReelAx8 : un manipulateur à 8 câbles et 6 ddl en configuration pleinement contrainte	13
1.11	CoGiRo : un manipulateur à 8 câbles et 6 ddl en configuration suspendue. Dans la seconde image, les câbles ont été repassés en blanc pour une meilleure visibilité. La plate-forme mobile est le cube central, ici avec les cibles (cinq ronds blancs) pour les expérimentations des outils de vision. Un porte palette peut y être fixé pour les expérimentations de palettisation.	14
1.12	Ensemble moteur, tambour et poulie de renvoi du robot <i>ReelAx8</i>	15
1.13	Ensemble actionneur linéaire, palan et poulies d'amplification	16
1.14	Exemple d'utilisation des moindres carrés basiques	25
1.15	Exemple d'utilisation de la régression des distances orthogonales	25
1.16	Exemple d'utilisation du contrôle de déviation.	26
1.17	Exemple de solution $\Sigma_{\exists, \exists}([A], [b])$	31
1.18	Enveloppe extérieure $[x]_{out}$ , boîte intérieure $[x]_{in}$ , pavage $\mathcal{L}_x$ et élément intérieur $x_{optim}$ optimisant le coût "le plus à droite".	32
2.1	Solutions $\square\Sigma_{\exists, \exists}$ obtenues avec les équations de Bjorck à une intersection de cercles épais : en haut lorsqu'ils s'intersectent, en bas lorsqu'ils ne s'intersectent pas.	46
2.2	Schéma du mécanisme de la barre.	49
2.3	Le mécanisme placé dans deux et trois configurations de mesure.	50
2.4	Vision globale des anneaux produits pour les trois jeux de mesures $(S_1, S_2, S_3)$	53
2.5	Résultats pour les boîtes extérieures en haut, pour les solutions ponctuelles en bas (représentées avec la meilleure boîte extérieure obtenue), avec le jeu $S_1$ à gauche et le jeu $S_2$ à droite.	57
2.6	Résultat du calcul de $[x]_{out}$ avec Newton-Bjorck-Hansen-Bliek pour le jeu $S_2$	58
2.7	Résultat du calcul de $[x]_{in}$ avec Newton-Bjorck-Beaumont pour le jeu $S_2$	59
3.1	Comment modéliser un câble dans le plan vertical contenant $A_i$ et $B_i$ ?	62
3.2	Un câble indéformable de masse nulle	63



3.3	Un câble élastique de masse nulle . . . . .	64
3.4	Un câble élastique de masse non nulle . . . . .	64
3.5	Un manipulateur parallèle à câbles . . . . .	66
3.6	L'encombrement du robot en pointillés, son espace de travail $\mathcal{W}_X$ et le plan diagonal $\mathcal{W}_B$ . . . . .	71
3.7	Un câble simplifié et l'erreur . . . . .	74
3.8	Le robot et le sous espace $\mathcal{SW}_X$ . . . . .	76
4.1	Tête du laser tracker . . . . .	89
4.2	Bases d'accueil de la cible du laser (entourées en rouge) . . . . .	90
4.3	La cible du laser . . . . .	90
4.4	Poses obtenues pour l'étalonnage du robot ReelAx8 . . . . .	91
4.5	Moyennes et écarts types des erreurs résiduelles du <i>MGI</i> avant et après auto-étalonnage en mm (moyennées sur les 30 poses d'étalonnage et les 8 câbles) . . . . .	93
4.6	Moyennes des erreurs résiduelles du <i>MGI</i> avant et après auto-étalonnage en mm (moyennées sur les 10 poses de validation et les 8 câbles) . . . . .	94
4.7	Erreur en positionnement % (avec les paramètres initiaux et après l'utilisation de 4 méthodes différentes) . . . . .	94
4.8	Erreur en orientation % (avec les paramètres initiaux et après l'utilisation de 4 méthodes différentes) . . . . .	95
4.9	Poses obtenues pour l'étalonnage du robot CoGiRo dans le sous espace de travail . . . . .	96
4.10	Poses choisies d'après l'expérience pour l'étalonnage du robot CoGiRo dans tout l'espace de travail . . . . .	96
4.11	Moyenne et écart type de la valeur du critère <i>ODR</i> avant et après étalonnage, avec des poses prises dans tout l'espace de travail $WS$ ou dans le sous espace, et donnés pour chacun des 8 câbles. . . . .	98
4.12	Moyenne et écart type de la valeur du critère $\chi^2$ avant et après auto-étalonnage, avec des poses prises dans tout l'espace de travail $WS$ ou dans le sous espace, et donnés pour chacun des 8 câbles . . . . .	99
4.13	Moyenne et écart type de l'erreur du <i>MGI</i> sur les poses de validation, en mm, avant et après étalonnage et auto-étalonnage et donnés pour chacun des 8 câbles. . . . .	101
5.1	Résultat sur l'incertitude finale du modèle géométrique pour chacun des huit câbles (en mètres) . . . . .	109
5.2	Intersection de boîtes dans le plan : sans et avec mesure aberrante . . . . .	117
5.3	Résultats (de gauche à droite et de haut en bas) pour $[a_1]$ , $[a_2]$ , $[a_4]$ , $[a_6]$ , $[a_7]$ et $[a_8]$ : boîte initiale puis du plus grand au plus petit $[a_i]_{pc}$ , $[a_i]_{pca}$ , $[a_i]_{cpc}$ . . . . .	120
5.4	Résultats pour $a_3$ et $a_5$ : boîte initiale et $[a_i]_{pc}$ , $[a_i]_{pca}$ , $[a_i]_{cpc}$ . . . . .	121
5.5	Résultats pour $a_3$ avec le système aberrant enlevé manuellement, la solution sans ce système est notée $[a_3]_{cpc2}$ . . . . .	121
5.6	Résultats des sous contracteurs pour $[a_3]_{cpc}$ , avec le résultat du sous système aberrant . . . . .	122
5.7	Résultats des sous contracteurs pour $[a_3]_{cpa}$ , avec le résultat du sous système aberrant . . . . .	122

# Liste des tableaux

3.1	Résultats (en gras) obtenus pour le robot ReelAx8 avec une méthode d'optimisation globale . . . . .	73
3.2	Résultats (en gras) obtenus pour le robot CoGiRo avec une méthode d'optimisation globale . . . . .	73
5.1	Enveloppe extérieure des paramètres après contraction avec la méthode Newton-Bjorck-Hansen-Bliek . . . . .	108
5.2	Ensemble intérieur tolérable des paramètres avec la méthode Newton-Bjorck-Beaumont . . . . .	109
5.3	Résultats du calcul de l'incertitude maximale de pose (position en cm et orientation en radians avec le formalisme d'Euler) avant et après étalonnage. . . . .	110
5.4	Résultats du calcul de l'erreur de modélisation en mm pour le câble reliant $A_1$ à $B_1$ et pour les 10 premières poses d'étalonnage. . . . .	114
5.5	Résultats de l'algorithme d'identification pour $\sigma = 0$ . . . . .	123
5.6	Résultats du calcul de l'incertitude maximale de pose (position en cm et orientation en radians avec le formalisme d'Euler) avant et après étalonnage. . . . .	123
B.1	Les paramètres géométriques théoriques du bâti du robot ReelAx8 . . . . .	147
B.2	Les paramètres géométriques théoriques de la plateforme du robot ReelAx8 . . . . .	147
C.1	Les paramètres géométriques théoriques du bâti du robot CoGiRo . . . . .	149
C.2	Les paramètres géométriques théoriques de la plateforme du robot CoGiRo . . . . .	149
D.1	Poses d'étalonnage et de validation du robot ReelAx8 (en mètres et en radians) . . . . .	152
D.2	Jeu de poses d'étalonnage du robot CoGiRo $S_1$ (en mètres et en radians) . . . . .	152
D.3	Jeu de poses d'étalonnage du robot CoGiRo $S_2$ (en mètres et en radians) . . . . .	152
D.4	Jeu de poses d'étalonnage du robot CoGiRo $S_3$ (en mètres et en radians) . . . . .	153
D.5	Jeu de poses d'étalonnage du robot CoGiRo $S_4$ (en mètres et en radians) . . . . .	153
D.6	Jeu de poses d'étalonnage du robot CoGiRo $S_5$ (en mètres et en radians) . . . . .	154



# Introduction générale

Le 20<sup>e</sup> siècle fut sans conteste celui de l'industrialisation et de l'informatique. Au carrefour de ces deux révolutions, la robotique s'est rapidement trouvée une place dans les chaînes de production. Les robots ont ainsi permis d'accélérer et d'améliorer la qualité des productions, tout en limitant les tâches répétitives et pénibles attribuées aux opérateurs humains - on connaît tous le film "les temps modernes" de Charlie Chaplin. Les robots manipulateurs d'architecture sérielle furent les premiers à envahir les chaînes de montage et de peinture dans la très florissante industrie automobile. Leur vocation étant de remplacer le bras humain, ils s'en inspirent naturellement.

Ces manipulateurs sériels atteignent leurs limites pour réaliser certaines tâches nécessitant, par exemple, une grande rigidité ou le déplacement d'objets lourds. Afin de pallier ces restrictions, une des solutions est d'utiliser une architecture de robot dite parallèle.

En effet, les robots manipulateurs d'architecture parallèle, c'est à dire dont plusieurs actionneurs manipulent ensemble et en parallèle le même organe terminal, possèdent des propriétés remarquables. Ils sont considérés comme rapides, précis, rigides et peuvent manipuler des charges lourdes. Ils furent en premier lieu utilisés comme simulateurs de vol, le plus connu de tous étant la plateforme de Gough-Stewart, une plateforme de simulation à six degrés de liberté. Malgré leurs bonnes performances, ils apparurent dans l'industrie seulement à partir des années 1980 avec le robot Delta du professeur Clavel. Ce dernier est d'abord entré en service dans le conditionnement à haute vitesse. Les robots parallèles ont toutefois souffert d'une architecture certes performante, mais difficile à concevoir et à contrôler et qui, de plus, limite l'espace de travail du manipulateur.

Récemment, une nouvelle classe de robot parallèle est apparue. Au lieu de jambes rigides, des câbles sont utilisés afin de fournir un espace de travail géométrique quasiment illimité et des performances dynamiques bien supérieures dues à la faible masse des câbles. Ils sont ainsi utilisés pour le survol des stades sportifs (spycam), l'orientation de télescope ou la manutention dans un parc logistique (déplacement et rangement de marchandises).

## Le projet

Cette thèse s'inscrit dans ce dernier cadre, au sein du projet CoGiRo : "Conception of Giant Robot", financé par l'ANR (Agence Nationale de la Recherche). Ce projet national s'est donné pour objectif de concevoir un robot à câbles de grande envergure afin d'automatiser et d'optimiser la gestion d'un parc logistique. Il comprend également la conception d'un réseau de caméras permettant une commande référencée vision du robot. Quatre partenaires sont impliqués dans le projet, l'entreprise Tecnalia et trois laboratoires de recherche : l'institut Blaise Pascal, le LIRMM et l'Inria.

## La problématique

La manipulation ou le rangement de marchandises nécessitent une bonne précision du robot, surtout dans un milieu où l'opérateur humain peut être présent. Pour

le projet CoGiRo, une précision de positionnement inférieure à 5cm est attendue. La tâche de déplacement des marchandises, ou palettisation, demande également une grande répétabilité du robot. La conception puis la construction du robot CoGiRo ont été réalisées avec cet objectif. Néanmoins, une étape d'amélioration de la précision est encore possible une fois le robot opérationnel, il s'agit de l'étalonnage.

L'étalonnage consiste en général à déterminer les valeurs réelles des paramètres du modèle servant à contrôler le robot. En ce qui nous concerne, les conditions pour étalonner le robot CoGiRo sont difficiles : la modélisation des robots à câbles est un sujet complexe, la précision visée est très importante par rapport aux dimensions de la structure et le robot a été conçu de manière à être peu sensible aux variations des paramètres de son modèle. De plus, afin de garantir un résultat fiable dans de telles conditions, il est nécessaire de définir de nouvelles méthodes pour réaliser l'identification des paramètres du robot, tout en considérant les nombreuses incertitudes en présence.

## Les objectifs

Les objectifs de cette thèse sont d'étalonner le robot afin de lui garantir une précision suffisante et de qualifier le robot pour une tâche de palettisation.

La suite de ce manuscrit est organisée de la façon suivante :

- Le chapitre 1 introduit les notions de base développées par la suite. Il présente les robots parallèles à câbles et leurs différentes architectures. Ensuite le principe de l'étalonnage est décrit et illustré par un état de l'art des approches et méthodes courantes. Enfin, l'analyse par intervalles, utilisée dans les chapitres suivants, est présentée dans une section préliminaire.
- Le chapitre 2 présente une approche de résolution des systèmes non linéaires sur-contraints d'intervalles. Deux méthodes sont ensuite appliquées et comparées dans le cas de l'étalonnage d'un mécanisme simple.
- Le chapitre 3 présente les modélisations géométrique et statique des robots à câbles, ainsi que la modélisation des câbles eux mêmes, importante source d'incertitude. Une méthode permettant de vérifier la validité de ces modèles au regard d'une approximation acceptable est proposée et illustrée sur deux prototypes.
- Le chapitre 4 montre l'étalonnage de ces deux prototypes utilisant une approche globale classique. Des méthodes nouvelles sont proposées afin de permettre un étalonnage performant malgré des conditions difficiles liées à la modélisation d'un robot à câbles et à l'échelle du robot étudié.
- Le chapitre 5 présente des approches reposant sur l'analyse par intervalles et qui permettent de garantir le résultat de l'étalonnage et de qualifier la précision du robot.

La conclusion de ce manuscrit récapitule les différentes contributions issues de cette thèse ainsi que les perspectives des travaux qui y sont liés.

# Notations et conventions

## Notations

Type	Exemple	Utilisation
Lettre minuscule	$l$	Scalaire
Lettre minuscule en gras	$\mathbf{a}$	Vecteur
Lettre majuscule en gras	$\mathbf{R}_{3 \times 3}$	Matrice de taille 3 par 3
Lettres minuscules entre crochets	$[x, y, z]$	Vecteur contenant les éléments $x$ , $y$ et $z$
Le signe $\times$	$\mathbf{v} \times \mathbf{w}$	Produit vectoriel de $\mathbf{v}$ par $\mathbf{w}$
Le signe $\ \cdot\ $	$\ \mathbf{v}\ $	Norme 2 du vecteur $\mathbf{v}$
Le signe $ \cdot $	$ x $	Valeur absolue de $x$
Lettre minuscule entre crochets	$[x]$	Intervalle $x$
Lettre minuscule en gras entre crochets	$[\mathbf{b}]$	Vecteur d'intervalles, ou boîte, $\mathbf{b}$
Lettre majuscule en gras entre crochets	$[\mathbf{A}]$	Matrice d'intervalles $\mathbf{A}$

## Conventions

Symbole	Utilisation
$\mathbf{a}_i$	Coordonnée du point de sortie du câble $i$ dans le repère du robot
$\mathbf{b}_i$	Coordonnée du point d'attache du câble $i$ dans le repère de la plateforme
$\boldsymbol{\rho}$	Variables articulaires
$\mathbf{x}$	Variables cartésiennes, définissent la pose de la plateforme dans le repère du robot
$\boldsymbol{\xi}$	Paramètres du robot
$m$	Nombre de câbles
$n$	Nombre de degrés de liberté
$\mathbf{p}$	Position de la plateforme dans le repère du robot
$\mathbf{R}$	Matrice de rotation de la plateforme par rapport au repère du robot
$l_i$	Longueur déroulée du câble $i$

## Les abréviations

Abréviations	Signification
ddl	Degré(s) de liberté
$MGD$	Modèle géométrique direct
$MGI$	Modèle géométrique inverse
$MGSD$	Modèle géométrico-statique direct
$MGSI$	Modèle géométrico-statique inverse



# Définitions et revue de littérature

## Sommaire

<b>1.1 Les manipulateurs</b>	<b>5</b>
1.1.1 Les manipulateurs parallèles	6
1.1.2 Les robots parallèles à câbles	8
<b>1.2 Amélioration de la précision</b>	<b>18</b>
1.2.1 Généralités sur l'étalonnage	20
1.2.2 Méthodes pour l'identification de paramètres	24
1.2.3 Dans le cadre de l'étalonnage	28
<b>1.3 Analyse par intervalles</b>	<b>28</b>
1.3.1 Arithmétique par intervalles	29
1.3.2 Schéma de Newton par intervalles	30
1.3.3 Systèmes linéaires d'intervalles	30
1.3.4 Caractérisation de la solution	32
1.3.5 Méthodes de traitement des systèmes d'intervalles	32
1.3.6 Contracteurs	35
<b>1.4 Conclusion</b>	<b>40</b>

## 1.1 Les manipulateurs

Les manipulateurs ont la fonction de positionner un solide, appelé l'organe terminal, par rapport à un autre solide de référence, appelé la base. Ce positionnement est réalisé suivant différents degrés de liberté (ddl). Il existe jusqu'à six ddl : trois en position (les translations) et trois en orientation (les rotations).

Deux grandes classes d'architecture mécanique existent pour réaliser un manipulateur.

La plus utilisée des architectures mécaniques est l'architecture sérielle (on parle alors de manipulateur sériel) composée d'une suite de solides reliés entre eux par des articulations, motorisées ou non, décomposables en articulations simples (1 ddl), rotation (articulation rotoïde) ou translation (articulation prismatique). Cette suite de solides forme ainsi une chaîne cinématique ouverte (voir figure 1.1(a)).

Ces manipulateurs s'inspirent de formes anthropomorphiques telles que le bras. Mais cette architecture n'est pas adaptée à toutes les tâches comme par exemple celles nécessitant une grande rigidité ou la manipulation d'objets lourds qui solliciterait fortement les articulations. Une des solutions est de répartir le poids de l'objet à manipuler sur plusieurs segments disposés en parallèle. Ce type d'architecture est celle utilisée par les manipulateurs parallèles. Nous nous intéressons plus particulièrement dans cette thèse à ce type d'architecture.



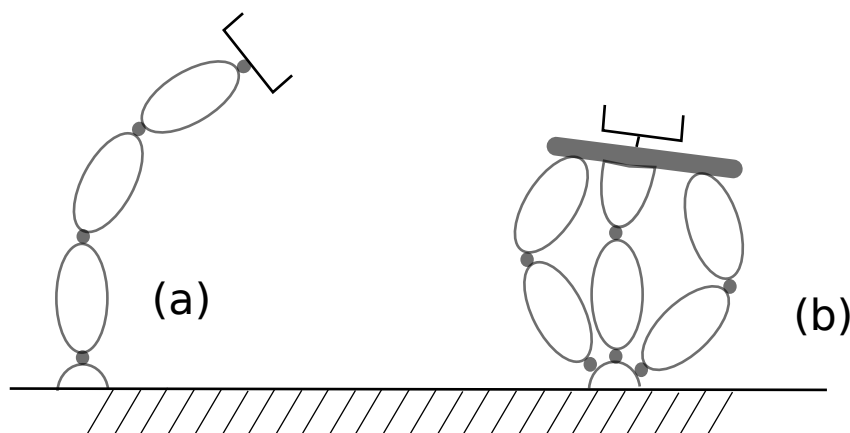


FIG. 1.1 – Architecture sériele (a) et parallèle (b) [Khalil 1999]

### 1.1.1 Les manipulateurs parallèles

#### 1.1.1.1 Définition

Une définition des robots parallèles, donnée dans [Merlet 1997], est : “Mécanisme en chaîne cinématique fermée dont l’organe terminal est relié à la base par plusieurs chaînes cinématiques indépendantes”. Une illustration en est donnée à la figure 1.1(b).

Pour un robot parallèle, l’organe terminal est plus fréquemment appelé plate-forme mobile. Nous utiliserons ce terme dans la suite du manuscrit.

#### 1.1.1.2 Exemples

L’histoire des robots parallèles est difficile à retracer, mais les travaux de [Merlet 2006] et de [Bonev 2003] ont permis de l’éclaircir. Il existe désormais de nombreux robots parallèles. Une liste conséquente des différentes architectures possibles est disponible sur le site de Jean-Pierre Merlet <sup>1</sup>.

Dans cette introduction on ne citera toutefois que deux robots parallèles, les plus marquants selon moi : la plate-forme de Gough et le robot Delta.

**La plate-forme de Gough :** La plate-forme de Gough (figure 1.2) est un robot parallèle à six degrés de liberté (trois en position et trois en orientation). Elle fut développée par Gough autour de 1950 pour tester l’usure des pneus. La plate-forme mobile, à laquelle une roue est solidement fixée, est reliée au sol par six segments dont la longueur varie. Chaque extrémité de ses segments, actionnés par des vérins pneumatiques, est connectée au sol par un cardan et à la plate-forme par une rotule. Les changements de longueur des vérins modifient la position et l’orientation de la plate-forme et donc de la roue, mais autorise aussi l’exercice d’une forte pression sur le pneu (ajustable en valeur et en direction).

La plate-forme de Gough (figure 1.2) tire de son architecture parallèle une grande rigidité.

<sup>1</sup><http://www-sop.inria.fr/members/Jean-Pierre.Merlet//merlet.html>

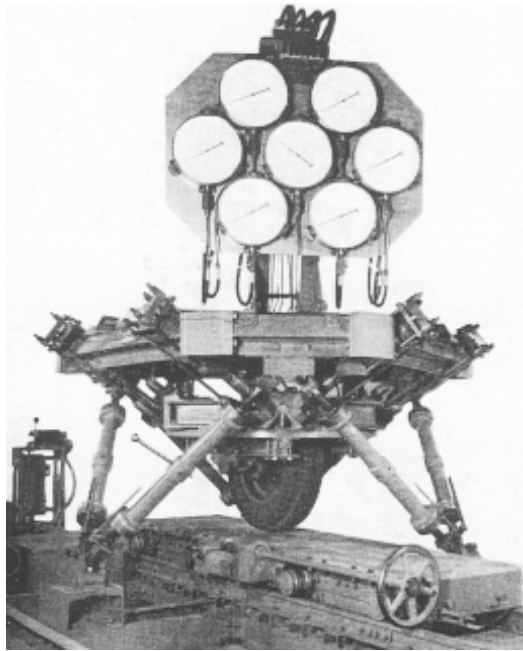


FIG. 1.2 – La plate-forme de Gough (Dunlop Tyre)

Ces caractéristiques font que la plate-forme de Gough est la structure utilisée pour les simulateurs de vol (figure 1.3) avec une dynamique intéressante pour recréer tous les mouvements ressentis dans une cabine de pilotage.

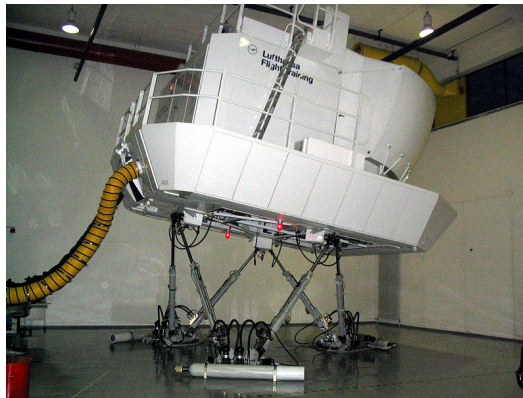


FIG. 1.3 – Un simulateur de vol basé sur l'architecture de la plate-forme de Gough-Stewart (Lufthansa)

**Le robot Delta :** Le robot Delta (figure 1.4) est un robot parallèle à trois degrés de liberté en translation, dont la structure mécanique à base de parallélogrammes permet théoriquement d'avoir une orientation constante.

Dans cette architecture, chaque moteur permet la rotation d'un levier, relié à

un parallélogramme par une liaison rotoïde, lui même relié à la plate-forme par une autre liaison rotoïde. Il fut inventé par Clavel en 1985 et trouva une application dans le conditionnement des boîtes de chocolat deux ans plus tard.



FIG. 1.4 – Un robot Delta industriel : le FlexPicker (ABB Flexible Automation)

Le robot Delta (voir figure 1.4) utilise les avantages de son architecture pour réaliser une tâche nécessitant précision et rapidité, le conditionnement à haute vitesse, pour laquelle il est devenu incontournable.

#### 1.1.1.3 Avantages et inconvénients

L'architecture parallèle tire partie de l'utilisation de plusieurs actionneurs travaillant conjointement et en parallèle à la même tâche et cela lui accorde [Merlet 1997] :

- une grande vitesse de manipulation ;
- une bonne précision ;
- la possibilité de manipuler des charges très lourdes ;
- une rigidité importante.

Toutefois, l'utilisation des robots parallèles implique certaines contraintes :

- un espace de travail de volume restreint ;
- des rotations limitées par les collisions entre les jambes et les singularités.

C'est entre autre afin de pallier certains de ces inconvénients que sont apparus les robots parallèles à câbles. Une description en est donnée à la section suivante.

### 1.1.2 Les robots parallèles à câbles

#### 1.1.2.1 Généralités et définition

Les robots parallèles à câbles sont une catégorie particulière de robots parallèles, dont les chaînes cinématiques ne sont pas des segments rigides mais des câbles. Ces câbles, attachés à la plate-forme mobile, sont chacun reliés à un actionneur

par différentes poulies de renvoi. L'actionneur permet le contrôle de la longueur de câble déroulée. Il peut être composé d'un moteur rotatif et d'un tambour, ou d'un actionneur linéaire couplé à un palan [Merlet 2008b]. Sur le même principe que les robots parallèles à actionneur prismatique, le changement de longueur des câbles autorise la commande de différents ddl de la plate-forme mobile.

On appelle *point de sortie* le point de contact du câble sur la dernière poulie ou le dernier œillet avant de s'attacher à la plate-forme. Le point de la plate-forme où est fixé le câble est appelé quant à lui le *point d'attache*. On nomme également le *bâti* (ou base) la partie rigide et fixe sur laquelle sont fixés les poulies et les actionneurs.

**Définition 1** *Un robot parallèle à câbles est un mécanisme en chaîne cinématique fermée dont la plate-forme mobile est reliée aux actionneurs au moyen exclusif de câbles.*

### 1.1.2.2 Historique

Les robots parallèles à câbles sont en pleine émergence et sont prometteurs pour répondre à certains problèmes que les robots parallèles à segments rigides ne parvenaient pas à résoudre. Par exemple, ce choix a été fait pour le futur télescope Fast [Kozak 2006]. Ce télescope de 500 mètres de diamètre n'aurait pas pu être manipulé par des segments rigides d'une telle longueur. L'utilisation des câbles devrait en faire l'un des plus grands et des plus précis des télescopes.

Toutefois, l'utilisation de câbles, cordes ou fils pour manipuler des objets dans l'espace est très ancienne.

Sous forme de grue, elle a servi aux égyptiens à construire les pyramides. Au moyen âge, les architectes utilisaient des cordes pour la construction des plus grands monuments (voir figure 1.5).



FIG. 1.5 – Une grue médiévale reconstruite

L'utilisation des fils pour manipuler un personnage a diverti la Birmanie dès le XV<sup>e</sup> siècle sous forme de marionnettes (voir figure 1.6).

Déjà dans ces applications, la grande légèreté des cordes ou des fils par rapport à leur robustesse ainsi que leur faible encombrement étaient vus comme des avantages.

Il est surprenant de découvrir à l'église du Sacré-Cœur de Menton, un vitrail représentant un paraplégique amené à Jésus de Nazareth au moyen de quatre câbles

FIG. 1.6 – Une marionnette birmane du XV<sup>e</sup> siècle

(voir figure 1.7). L'anecdote est d'autant plus intéressante que cette application est proche de celle qui aujourd'hui motive l'équipe Coprin avec les manipulateurs Marionet-Crane et Marionet-Assist : le transfert de personnes invalides.



FIG. 1.7 – Vitrail de l'église du Sacré-Cœur de Menton

Plus récemment et dans un tout autre registre, le monde du cinéma a trouvé dans les dispositifs actionnés par des câbles le moyen d'obtenir des angles de vue spectaculaires. Ainsi pour le film "L'Argent" tourné en 1928, les opérateurs ont pu positionner la caméra pratiquement n'importe où dans la scène, avec seulement trois câbles, donnant libre court à l'imagination du réalisateur, Marcel L'Herbier.

Ce film est devenu par la suite célèbre justement pour ses mouvements de caméra, lui conférant un dynamisme et une subjectivité surprenants à l'époque.

L'idée la plus ancienne n'a donc sans doute pas été de remplacer les jambes rigides d'un manipulateur parallèle par des câbles mais d'utiliser plusieurs câbles pour manipuler un objet. Le cheminement intellectuel a donc été le même que celui qui a amené à considérer l'architecture des manipulateurs parallèles après les sériels : porter des charges plus lourdes, augmenter les degrés de liberté, obtenir une meilleure rigidité.

### 1.1.2.3 Deux types de configuration pour les robots à câbles

On peut distinguer deux types de configurations pour les manipulateurs parallèles à câbles, ceux en configuration grue ou suspendue et ceux en configuration pleinement contrainte <sup>2</sup>.

**Configuration suspendue :** La configuration suspendue peut être définie par la position des points de sortie des câbles. Dans cette configuration, ils sont au dessus de l'espace de travail du robot, de telle sorte que la plate-forme est suspendue par les câbles et qu'aucun câble ne peut exercer une force verticale vers le bas : la gravité joue un rôle essentiel pour l'équilibre mécanique du robot (figure 1.8) [Carricato 2012b]. Contrairement aux manipulateurs parallèles classiques, munies de jambes rigides, ici le caractère unidirectionnel de l'actionnement par câbles (un câble peut "tirer" mais ne peut pas "pousser") peut empêcher le contrôle de tous les ddl, même avec plus de câbles que de ddl.

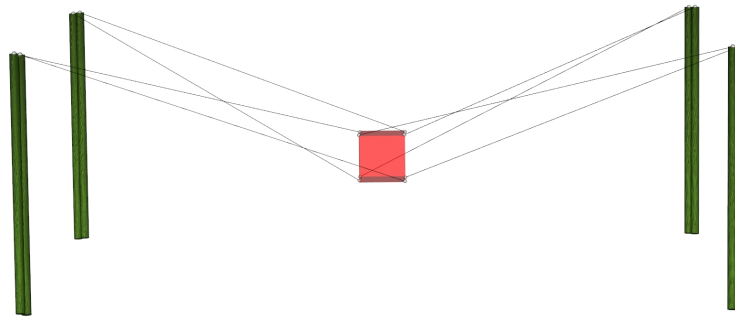


FIG. 1.8 – Manipulateur à 8 câbles et 6 ddl en configuration suspendue

Les robots Fast [Kozak 2006], Nims [Borgstrom 2009b], Skycam [sky 2007], ou encore le Nist [Albus 1993] sont des exemples remarquables de robots parallèles à câbles en configuration suspendue.

**Configuration pleinement contrainte :** Dans cette configuration, l'espace de travail est inclus dans le polyèdre formé par les points de sortie des câbles. Les

<sup>2</sup>Les images sont issues d'un outil de visualisation/simulation/étalonnage basé sur vtk/gsl et développé par l'auteur.

câbles se trouvent ainsi de part et d'autre de la plate-forme. Dans une configuration pleinement contrainte, les câbles travaillent en opposition (figure 1.9).

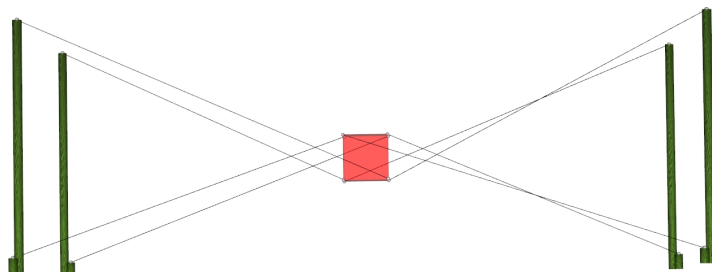


FIG. 1.9 – Manipulateur à 8 câbles et 6 ddl en configuration pleinement contrainte

Pour cette catégorie on peut citer les robots Marionet-Rehab [Merlet 2008b], ou Falcon [Kawamura 1995, Kawamura 2000].

#### 1.1.2.4 Les robots étudiés dans le cadre du projet CoGiRo

**Le prototype ReelAx8 :** Le robot reelAx8 (figure 1.10) étudié dans les travaux présentés dans ce manuscrit fait partie de la catégorie des robots pleinement contraints (mais il peut aussi être changé en configuration suspendue [Lamaury 2012]). Il sert de prototype au robot CoGiRo et permet de vérifier et tester une partie des méthodes [Lamaury 2013, Dallej 2011]. Ses caractéristiques sont données en annexe B.

**Le robot CoGiRo :** Le robot CoGiRo (figure 1.11) étudié dans les travaux présentés dans ce manuscrit entre dans la catégorie des manipulateurs parallèles en configuration suspendue.

Il s'agit du principal robot de cette étude, l'architecture suspendue a été choisie en vue d'une tâche de transfert d'objet (pick & place). Certaines approches ont été adaptées à ce robot de grande échelle [Dallej 2012, Riehl 2011]. Ses caractéristiques sont données en Annexe C.

#### 1.1.2.5 Les travaux sur les robots à câbles

Un nombre conséquent de travaux menés sur les robots parallèles à câbles ont été publiés.

**Les premières études :** Un manipulateur parallèle où des câbles remplacent les segments rigides a été proposé dans [Landsberger 1985]. Dans [Ming 1994], il a été établi que pour qu'une architecture soit suffisamment contrainte, il fallait au minimum  $n + 1$  câbles pour  $n$  ddl.

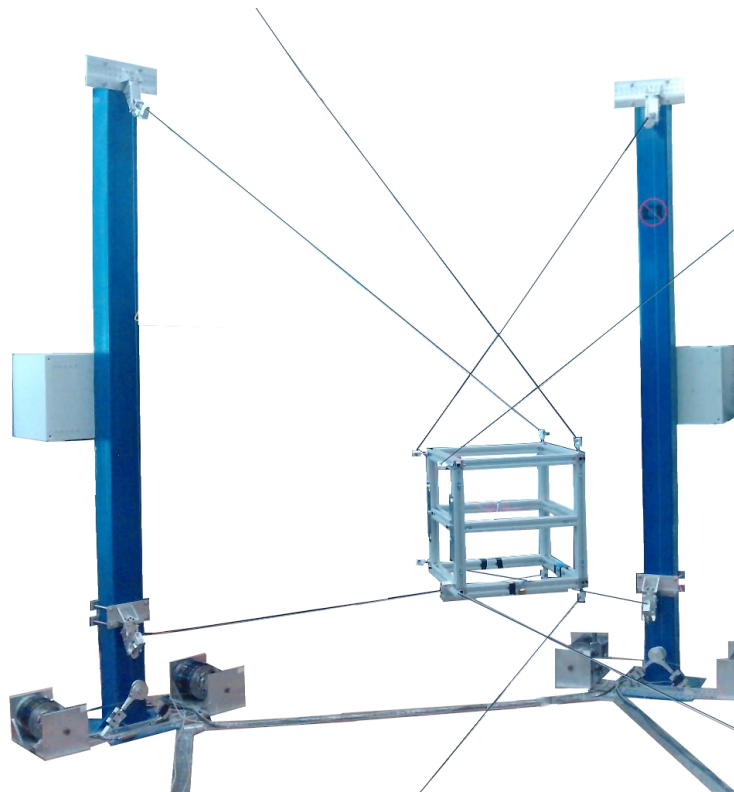


FIG. 1.10 – ReelAx8 : un manipulateur à 8 câbles et 6 ddl en configuration pleinement contrainte





FIG. 1.11 – CoGiRo : un manipulateur à 8 câbles et 6 ddl en configuration suspendue. Dans la seconde image, les câbles ont été repassés en blanc pour une meilleure visibilité. La plate-forme mobile est le cube central, ici avec les cibles (cinq ronds blancs) pour les expérimentations des outils de vision. Un porte palette peut y être fixé pour les expérimentations de palettisation.

**Quelques applications :** Dès lors, différentes conceptions et applications de robots à câbles ont été proposées dans la littérature. Des prototypes de robots ultra-rapides [Kawamura 1995], des manipulateurs à quatre ddl [Kosowski 2002], des robots permettant l'acquisition d'images 3D d'un objet [Bouchard 2008], jusqu'aux interfaces haptiques [Zitzewitz 2009]. La société Haption commercialise un exemple d'interface haptique à 6ddl reposant sur un robot parallèle à câbles, le robot *Inca 6D*. Ce dernier est adapté du manipulateur *Spidar*, "Space interface device for artificial reality", du Professeur Sato [Sato 2002].

Depuis quelques années, de nombreux prototypes sont imaginés et construits dans les laboratoires du monde entier [Williams 2003, Ogahara 2003, Ferraresi 2004, Pusey 2004, Behzadipour 2005, Oh 2005a, Dekker 2006, Borgstrom 2009a].

**La mécanique :** La longueur des câbles est classiquement contrôlée par l'utilisation d'un tambour actionné par un moteur rotatif (voir figure 1.12). Certains tambours sont équipés d'un guide (c'est le cas pour *ReelAx8* et *CoGiRo*). Avec une translation couplée à la rotation de l'enrouleur (en général par une vis sans fin), le guide se déplace parallèlement au tambour et conduit le câble tout au long de son enroulement ou déroulement. Un tambour avec guide permet d'enrouler une longueur de câble identique à chaque tour, contrairement à un tambour sans guide où le fil est libre de se superposer au fur et à mesure de l'enroulement. Le guide évite donc la superposition, ce qui élimine l'effet de cisaillement du câble sur lui même. Cependant, si un tambour avec guide possède certains avantages, sa conception est plus complexe et son coût de fabrication est plus important qu'un tambour sans guide.

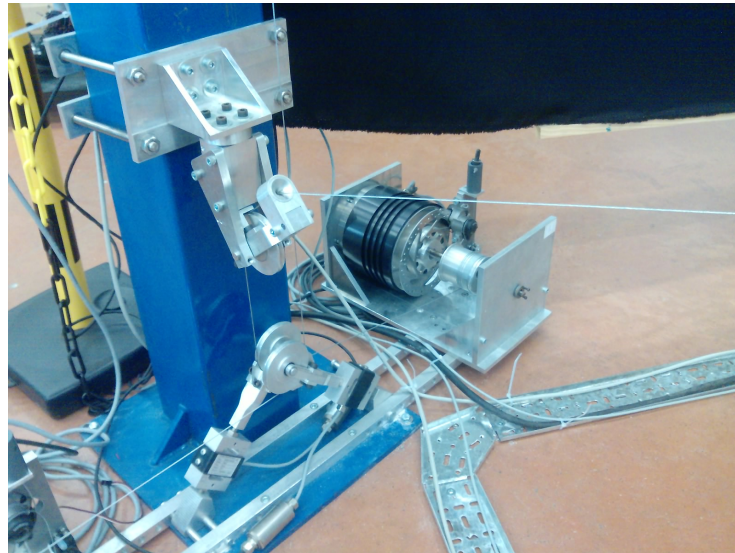


FIG. 1.12 – Ensemble moteur, tambour et poulie de renvoi du robot *ReelAx8*

Un actionnement linéaire couplé à un palan est également possible et offre à la plate-forme mobile un comportement intéressant [Merlet 2008b]. L'utilisation d'un palan, composé de plusieurs poulies, peut offrir une plus grande précision qu'un tambour tout en amplifiant les déplacements (voir figure 1.13). Avec de tels actionneurs,

les manipulateurs à câbles peuvent, en théorie, atteindre des vitesses très élevées, principalement en configuration pleinement contrainte.

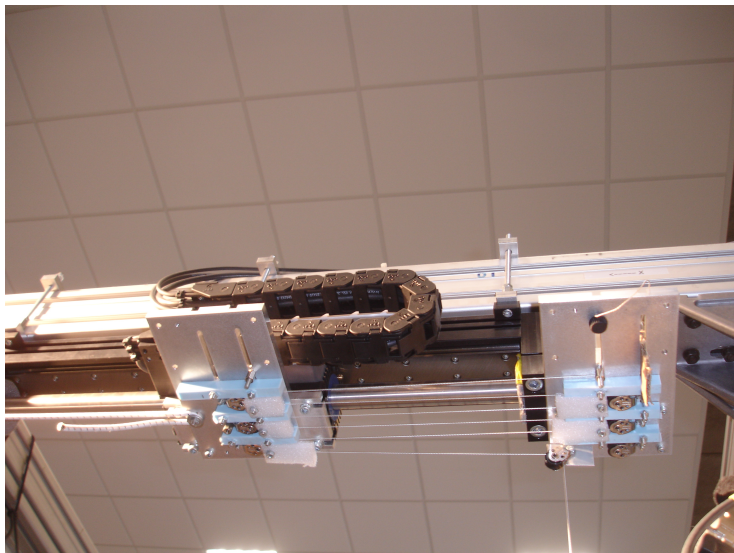


FIG. 1.13 – Ensemble actionneur linéaire, palan et poulies d'amplification

**La modélisation des câbles :** Les câbles sous tension peuvent être considérés comme des segments rigides afin de modéliser le manipulateur comme un robot parallèle classique [Tsai 1996, Roberts 1998]. La modélisation des câbles, leur raideur et leur élasticité, est en effet un problème en soi (en particulier pour les câbles synthétiques dont le comportement physique, élasticité ou déformation plastique par exemple, diffère sensiblement de celui des câbles en acier, plus largement étudiés). Leurs effets sur le modèle géométrique ou sur la statique ne sont en général pas négligeables [Kozak 2006, Behzadipour 2006, Merlet 2009, Riehl 2009]. Cependant, les câbles n'étant pas rigides, ils ne peuvent exercer une force bilatérale. Cette contrainte complexifie fortement l'analyse de ce type de manipulateur.

**Les forces :** En configuration pleinement contrainte, la tension peut croître jusqu'à la rupture des câbles ou la détérioration du robot. En plus de la longueur des câbles, la tension se doit donc d'être surveillée [Lim 2009]. Un modèle géométrico-statique a été introduit dans [Carricato 2012b]. Il consiste dans la résolution simultanée de la géométrie et de la statique, intrinsèquement liée, au moyen d'un algorithme qui prend aussi en compte la stabilité de l'équilibre statique. Le modèle géométrico-statique inverse d'un robot à quatre câbles en configuration suspendue, consistant à trouver toutes les positions d'équilibre de la plate-forme pour une position du centre de masse donnée, a été résolu dans [Carricato 2012a]. Ceci a nécessité la résolution d'un polynôme univarié de degré 32 (obtenu après un lourd travail d'élimination) au moyen des bases de Gröebner. La résolution du modèle géométrico-statique inverse est donc un problème compliqué avec quatre câbles (pourtant considérés sans masse et sans élasticité).

Et enfin, le modèle géométrico-statique direct d'un robot à trois câbles en configuration suspendue, consistant à trouver toutes les poses de la plate-forme en fonction de la longueur des câbles, a été résolu avec l'analyse par intervalles dans [Berti 2013]. Déterminer la position de la plate-forme en fonction de la longueur des câbles pour les robots à quatre ou cinq câbles en configuration suspendue (sans mesures de la tension dans les câbles) est encore un problème ouvert [Merlet 2012b].

**La dynamique :** Les modèles dynamiques des manipulateurs à câbles sont assez similaires à ceux des robots parallèles classiques [Bruckmann 2008b], et l'emploi des câbles peut même servir à augmenter l'espace de travail du robot [Barrette 2005]. [Oh 2005b, Gosselin 2012] proposent des commandes dynamiques.

**L'espace de travail :** Un des apports des câbles aux manipulateurs parallèles est le fait qu'ils permettent un espace de travail qui est moins limité par les débattements des actionneurs comme les robots parallèles classiques [Landsberger 1992, Verhoeven 2000]. En effet, il est en théorie possible de créer un espace de travail maximal très vaste en utilisant de grandes longueurs de câbles. Cependant, l'espace de travail d'un manipulateur parallèle à câbles doit, comme un robot parallèle classique, satisfaire aux contraintes géométriques, mais doit également conserver une tension positive dans les câbles [Stump 2006, Gouttefarde 2006]. Cet espace de travail peut être formulé sous différentes formes [Lau 2011, McColl 2011].

**Les collisions :** Avec un nombre de câbles important, des collisions peuvent se produire entre les câbles ou entre un câble et la plate-forme. Ce phénomène a été étudié afin de définir un espace de travail sans collisions [Merlet 2004a].

**La commande :** Dans le cadre du projet CoGiRo, le robot dispose d'une commande référencée vision dédiée aux robots à câbles afin d'offrir une commande précise [Dallej 2012]. La maniabilité d'un manipulateur à câbles peut être importante et être utilisée dans une application haptique [Gallina 2002]. Afin d'attribuer des tâches au robot, il peut être utile de lui définir une trajectoire tout en considérant l'élasticité des câbles pour permettre un déplacement sans à-coups [Agahi 2010].

**Les singularités :** Si l'analyse des singularités des robots parallèles classiques est une étape importante afin d'éviter leur détérioration, celle des robots à câbles est un problème qui a peu été étudié [Merlet 2012b]. L'analyse des singularités des robots parallèles à câbles est jusqu'à présent considérée comme équivalente à celle des robots parallèles classiques [Ottaviano 2006]. Si les singularités classiques sont présentes et peuvent avoir éventuellement les mêmes conséquences (les tensions dans les câbles qui tendent vers l'infini), une autre classe de singularités peut être définie pour une pose où plusieurs configurations sont possibles pour les câbles tendus [Merlet 2012b].

Néanmoins, une analyse des singularités en fonction de la raideur des câbles est proposée dans [Verhoeven 1998].

**L'étalonnage :** L'étude des manipulateurs parallèles à câbles passe également par la phase d'étalonnage. Nous verrons l'étalonnage particulier des robots à câbles à la section 1.2.1.4.

### 1.1.2.6 Avantages et inconvénients

Les manipulateurs parallèles à câbles ont de nombreux avantages, issus de l'architecture parallèle et de l'utilisation des câbles :

- espace de travail plus important que celui des robots parallèles en raison des débattements d'actionnement plus importants ;
- mécanique simplifiée par la suppression des liaisons passives (rotoïde et sphérique) ;
- déport possible des actionneurs en dehors de l'espace de travail ;
- facilement reconfigurable ;
- faible encombrement des segments ;
- bas coût ;
- légèreté des corps en mouvement donc faible inertie et grande dynamique.

Cependant, la prise en compte des caractéristiques physiques des câbles peut rendre la modélisation du robot complexe. L'hypothèse visant à les considérer comme des segments rigides, sans masse et non déformables est souvent faite. Cette simplification peut ajouter des incertitudes que nous considérerons au chapitre 3.

Avec une modélisation réaliste des câbles, le modèle géométrique inverse (*MGI*), qui consiste à calculer les longueurs de câbles ( $l_i$ ) à dérouler pour atteindre une position ( $\mathbf{p}$ ) et une orientation ( $\mathbf{R}$ ) données, est plus complexe que celui des robots parallèles classiques. En effet, la longueur des câbles effectives dépend de la tension dans les câbles à cause de l'élasticité et de la pesance des câbles, et elle est donc fonction de la statique. Lorsque ce phénomène est pris en compte dans la modélisation inverse du manipulateur, on parle alors de modèle géométrico-statique inverse [Carricato 2012a]. Ceci est un problème très complexe et encore très ouvert pour une modélisation réaliste des câbles.

Le modèle géométrique direct (*MGD*), qui consiste à calculer la pose du robot (position  $\mathbf{p}$  et orientation  $\mathbf{R}$  de la plate-forme) en fonction des longueurs de câble déroulées, dépend aussi de la statique. Le modèle géométrico-statique direct est là encore un problème ouvert [Merlet 2012b].

## 1.2 Amélioration de la précision

L'attribution d'une tâche à un robot manipulateur consiste en général à lui définir une trajectoire à réaliser ou une série de poses à atteindre. Ces poses ou ces trajectoires sont spécifiées dans l'espace généralisé alors que la commande s'effectue dans l'espace articulaire, ce qui nécessite la définition des modèles du robot explicitant les relations entre l'espace généralisé et l'espace articulaire. Ces modèles ont donc une importance très grande sur les performances du manipulateur. Cependant, le robot est un mécanisme physique soumis à des phénomènes pouvant ne pas être complètement maîtrisés et entraînant des incertitudes sur son comportement.

On peut en distinguer deux origines [Siciliano 2008, Joubair 2012] :

- Les incertitudes d'origine géométrique sont liés à la précision du modèle cinématique du robot : les imprécisions d'assemblage ou de fabrication et les biais articulaires.
- Les incertitudes d'origine non-géométrique sont attribués à la qualité des composantes mécaniques du robot : les déformations des éléments mécaniques dues aux efforts, aux variations de température ou bien encore aux jeux mécaniques dans les articulations.



Les modèles d'un robot ne considèrent pas toutes ces sources d'incertitude car ils sont issus d'un compromis entre simplicité et représentation réaliste du comportement du robot, connaissances théoriques et informations disponibles pour le renseigner (les données des capteurs).

Les trois principales origines d'incertitude sur le comportement d'un mécanisme peuvent être illustrées par un exemple. Prenons une barre de longueur  $l$  tournant autour d'un axe d'un angle  $\theta$  mesuré par un capteur. La position de l'extrémité de la barre décrit un cercle.

- La longueur  $l$  n'est pas connue exactement mais estimée d'après la phase de fabrication. Il s'agit d'une incertitude de type géométrique (ou incertitude sur les paramètres).
- L'axe a un peu de jeu à cause de l'usinage. Il s'agit d'une incertitude non-géométrique. Ce type d'incertitude est très complexe à modéliser.
- La barre n'est pas de rigidité infinie et l'extrémité de la barre peut ne pas décrire un cercle si elle est soumise à des efforts externes. Si la barre est considérée comme rigide dans les équations de modélisation, une erreur est alors engendrée. On parle alors d'une erreur de modèle ou d'une incertitude due à la modélisation.

On peut ajouter que le capteur est également imprécis et ne fournit pas  $\theta$  mais  $\theta \pm \varepsilon$ . Il s'agit d'une incertitude de mesure.

Toutes ces incertitudes jouent un rôle sur la précision du manipulateur [Wang 1993]. Par précision, il faut comprendre exactitude de positionnement et d'orientation (de pose) ainsi que la répétabilité. La norme *ISO 9283 (Robots manipulateurs industriels : critères de performance et méthodes d'essai correspondantes)* donne une définition de l'exactitude et de la répétabilité de pose afin de caractériser les performances d'un manipulateur :

- Exactitude de pose : exprime l'écart entre une pose commandée et la moyenne des poses atteintes. Elle comprend l'exactitude de positionnement (la différence entre la position d'une pose commandée et le barycentre des positions atteintes) et l'exactitude d'orientation (la différence entre l'orientation d'une pose commandée et la moyenne des orientations atteintes) [Brethé 2012].
- Répétabilité de pose : exprime l'écart entre les poses atteintes pour la même pose commandée et répétée  $N$  fois. Elle s'exprime par la répétabilité de positionnement (le rayon de la sphère dont le centre est le barycentre du nuage de points et qui contient les  $N$  poses atteintes) et la répétabilité d'orientation (l'étendue des variations angulaires autour des coordonnées angulaires moyennes) [Brethé 2012].

Si l'on souhaite améliorer la précision d'un manipulateur, les incertitudes sur la modélisation, sur les paramètres de cette modélisation et sur les mesures des capteurs renseignant cette modélisation doivent être considérées ou diminuées. Ce travail peut se faire tout au long du processus conduisant à l'élaboration d'un manipulateur opérationnel :

- En phase de conception :
  - Chercher une conception de robot qui minimise l'effet des incertitudes de mesure articulaire sur la précision ;
  - Prévoir l'ajout de capteurs proprioceptifs afin de mesurer l'état du robot ;
  - Étudier le robot et définir un modèle adéquat (qui considère les incertitudes les plus importantes) ;
- En phase de construction :

- Utiliser des capteurs de précision ;
- Améliorer la qualité de fabrication et d'assemblage ;
- À partir d'un robot opérationnel :
  - Étalonner le robot, c'est à dire identifier les paramètres géométriques qui définissent les modèles du robot (généralement le modèle utilisé par la commande) ;

Nous verrons au chapitre 4 comment on a cherché à améliorer la précision du robot CoGiRo (voir figure 1.11) tout au long de son processus de conception et de fabrication.

### 1.2.1 Généralités sur l'étalonnage

L'étalonnage géométrique consiste à améliorer la connaissance des paramètres qui définissent la géométrie du robot et les éventuels biais des actionneurs.

L'approche la plus naturelle consiste à déterminer l'erreur entre l'observation (la mesure) de la position et de l'orientation de l'organe terminal et sa prédiction, déterminée avec les modèles du robot. L'identification consiste alors à déterminer les paramètres du modèle tels que cette erreur soit la plus faible possible. En d'autres termes, cela revient à identifier les paramètres rentrant en jeu dans la commande de sorte à minimiser l'erreur de positionnement de l'organe terminal [Everett 1987].

Cette approche est réductrice et mal adaptée à l'étalonnage des robots parallèles [Daney 2000]. En effet, pour les robots parallèles le calcul de l'erreur de positionnement nécessite la résolution d'un système d'équations (voir section 1.2.1.5). On préfère la vision plus générale de [Wampler 1995] qui unifie les différentes méthodes d'étalonnage que nous verrons ci après (étalonnage, auto-étalonnage et étalonnage avec contraintes) et qui considère des équations de fermetures.

De nombreux travaux ont été menés sur l'étalonnage des robots parallèles [Vischer 1996, Daney 2000, Besnard 2000, Renaud 2003, Deblaise 2006, Fazenda 2006, Corbel 2008].

L'étalonnage géométrique est une étape importante car les incertitudes sur les paramètres géométriques sont ceux qui influencent le plus la précision des manipulateurs parallèles [Wang 1993]. Toutefois, à l'opposé des manipulateurs sériels, cette influence est limitée pour une architecture parallèle.

De plus, une mauvaise estimation de ces paramètres peut impliquer l'apparition de comportements imprévus comme des singularités ou des collisions internes dans un espace de travail en théorie sûr.

En croisant la littérature sur l'étalonnage, il est possible d'en faire ressortir les grandes étapes :

1. Choisir un modèle pour le robot (en général celui utilisé par la commande) ;
2. Ajouter éventuellement des capteurs ou des contraintes mécaniques afin d'obtenir des informations redondantes sur l'état interne du robot ;
3. Modéliser les équations d'étalonnage ;
4. Étudier l'identifiabilité théorique des paramètres de ces équations ;
5. Chercher des configurations où réaliser les mesures pour un étalonnage performant ;
6. Acquérir des mesures ;
7. Vérifier l'identifiabilité expérimentale des paramètres ;

8. Identifier des paramètres ;
9. Valider les résultats de l'identification.

Dans le chapitre 4, chacune de ces étapes est développée sur les robots étudiés et certaines adaptées à l'étalonnage particulier des robots parallèles à câbles.

Si nous avons opté pour la vision unifiée de l'étalonnage proposée par [Wampler 1995], la pratique conduit à distinguer trois types d'étalonnage, suivant la provenance des mesures : l'étalonnage, l'auto-étalonnage et l'étalonnage avec contraintes.

### 1.2.1.1 Étalonnage

On parle d'étalonnage si les mesures ont nécessité l'utilisation d'appareils de mesure externe en plus des capteurs proprioceptifs (entrant en jeu dans la commande).

La mesure additionnelle la plus naturelle à obtenir est celle de la pose (totale ou partielle) de l'organe terminal. Pour cela on peut mesurer la pose avec un théodolite [Zhuang 1995] ou une caméra [Renaud 2005]. La mesure de la pose complète de l'organe terminal peut être réalisée avec un laser tracker [Meng 2003]. Cependant, il est possible de mesurer également une distance entre la base et l'organe terminal [Takeda 2004], ou de se contenter d'une information partielle comme l'orientation de l'organe terminal [Besnard 1999].

Dans ces premières études, seul l'aspect géométrique était considéré dans l'étalonnage, en exploitant par exemple la simplicité des équations géométriques du robot Delta dans [Vischer 1998]. Depuis peu, l'étalonnage considérant plusieurs aspects physiques comme la déformation due à la température et utilisant donc un modèle très complexe du manipulateur a pu être étudié dans [Lubrano 2011].

Les méthodes d'étalonnage nécessitent d'identifier les paramètres supplémentaires induits par le besoin de lier le dispositif de mesure externe au repère de référence du robot [Everett 1987].

### 1.2.1.2 Auto-étalonnage

On parle d'auto-étalonnage si les mesures proviennent exclusivement des capteurs proprioceptifs du robot.

Cela est possible s'il y a redondance de capteurs [Zhuang 1998, Patel 2000]. C'est à dire si le robot fournit de lui même plus d'information sur son état qu'il n'en nécessite pour le déterminer. Ainsi ce surplus d'information peut être utilisé pour auto-étalonner le robot. On parle alors de redondance d'informations ou métrologique [Marquet 2002] (différente de la redondance d'actionnement).

La redondance d'informations obtenue sur une plate-forme de Gough en sur-instrumentant d'origine le manipulateur (capteurs d'angles sur les cardans de trois segments sur les six) a permis non seulement d'auto-étalonner le robot, mais aussi de simplifier le *MGD* [Daney 2000].

Une approche consistant à considérer les chaînes cinématiques fermées d'un robot parallèle est envisageable pour auto-étalonner celui-ci [Iurascu 2003].

Dans le chapitre 3, nous verrons que l'auto-étalonnage des robots à  $m > n$  câbles n'est possible que sous certaines hypothèses.



### 1.2.1.3 Étalonnage avec contraintes

Avec l'ajout de contraintes mécaniques, on peut limiter les ddl du manipulateur et ainsi rendre les capteurs internes redondants.

En effet, il est possible de contraindre de manière passive l'organe terminal d'un manipulateur et d'exploiter les chaînes cinématiques fermées ainsi obtenues pour étalonner le mécanisme [Bennet 1991, Maurine 1996].

En bloquant la longueur et la direction d'un segment d'une plate-forme de Gough, [Daney 1999, Daney 2000] montre que l'on simplifie les équations d'étalonnage, ce qui permet un résultat moins sensible au bruit de mesure. Il montre également que les contraintes mécaniques fournissent un moyen de se passer de mesures extéroceptives et qu'ainsi on se ramène à un problème d'auto-étalonnage. En effet, en limitant les ddl de l'organe terminal, les capteurs proprioceptifs fournissent alors la redondance de mesure nécessaire à l'auto-étalonnage.

### 1.2.1.4 Cas particulier des robots parallèles à câbles

Plusieurs solutions ont été proposées pour les robots parallèles, en particulier certaines d'entre elles peuvent être adaptées à l'étalonnage des manipulateurs à câbles. Dans [Patel 2000, Takeda 2004], un capteur est employé pour fournir une mesure de distance additionnelle d'un point de la base à un point de la plate-forme mobile d'une plate-forme de Gough-Stewart. Partant de ce principe et dans le cas d'un robot parallèle à 6 ddl et 8 câbles, et sous certaines hypothèses, 2 des 8 câbles peuvent être considérés comme des capteurs additionnels. Nous verrons au chapitre 4 que cette approche nécessite de négliger l'élasticité et la masse des câbles et qu'elle suppose également que les longueurs déroulées (définies par la commande) approximent suffisamment les distances entre les sorties de câbles et les points d'attache.

Une approche est proposée pour l'étalonnage géométrique des robots reconfigurables par [Tadokoro 1999]. Pour cela, un cube est positionné dans l'espace de travail du robot, il sert de contrainte mécanique. Lors de l'étalonnage, la plate-forme est positionnée sur chaque sommet de ce cube, les câbles sont tendus et la mesure des longueurs de câbles est effectuée. Ces mesures permettent ensuite l'identification de la position des treuils du robot.

Une simulation de l'étalonnage géométrique d'un robot hybride à quatre ddl composé de 3 câbles et d'un mécanisme fait d'un parallélogramme est décrit dans [Varziri 2007]. Ce robot, d'architecture complexe, est étalonné en 2 temps, le mécanisme articulé, puis le manipulateur à câbles.

Le robot Nims-PL, un robot plan à 4 câbles et 2 ddl, est auto-étalonné au moyen de 2 méthodes dans [Borgstrom 2009b]. Une approche utilise les mesures de la tension dans les câbles au moyen de capteurs haute résolution. Cependant, le coût d'un appareil de mesure de tension adapté est élevé, sa plage d'utilisation optimale (avec une tolérance acceptable) est limitée et les mesures obtenues sont parfois trop bruitées pour être utilisables (nous verrons que c'est le cas avec le robot CoGiRo). Une seconde méthode permet également la redondance d'information en utilisant de petits changements sur la longueur d'un câble après l'autre et en mesurant l'effet sur la longueur des câbles opposés.

Récemment, l'auto-étalonnage d'un robot pleinement contraint à 8 câbles pour 6 ddl (similaire au prototype *ReelAx8*) a été réalisé en utilisant les informations de longueur et de tension dans les câbles [Miermeister 2012]. L'influence des bruits de

mesure des capteurs d'effort est également démontrée et confirme la nécessité de disposer des mesures de tension de très bonne qualité pour obtenir un étalonnage efficace.

### 1.2.1.5 Modèles d'étalonnage

Une fois les mesures obtenues par l'une des méthodes évoquées ci dessus, on se doit de les lier avec les paramètres géométriques au travers des modèles du robot. Il peut être nécessaire d'introduire des variables supplémentaires lors de la modélisation si les informations sur la pose du robot ne sont que partielles (par exemple si l'on mesure uniquement la position de l'organe terminal). Les variables restantes (par exemple l'orientation) devront être soit éliminées [Daney 2000], soit identifiées. Ainsi, éliminer les variables de pose constitue une des difficultés majeures de l'auto-étalonnage des robots parallèles (voir section 4.1.2).

Avec une approche reposant sur l'expérience des manipulateurs sériels, les premiers étalonnages de robots parallèles utilisèrent le *MGD* et donc cherchèrent à identifier les paramètres géométriques du robot tels que la prédiction de pose donnée par le modèle direct soit la plus proche possible de la mesure effectuée.

Si l'utilisation de la mesure de pose est directe avec cette approche, la résolution du *MGD* entraîne des problèmes (existence de solution, convergence [Merlet 2006]). L'utilisation du *MGI* au contraire simplifie l'identification en découplant les équations segment par segment. On cherche alors à identifier les paramètres géométriques du robot tels que la prédiction de la longueur du segment donnée par le modèle inverse soit la plus proche possible de la mesure effectuée.

Une vision unifiée de l'étalonnage est donnée par [Wampler 1995]. Elle consiste à voir toute mesure ou contrainte mécanique comme une information permettant d'instancier une équation implicite. L'étape d'identification va donc consister à trouver les paramètres tels que toutes ces équations tendent à être satisfaites.

Dans cette approche [Wampler 1992], pour chacune des  $N_C$  mesures de pose, les équations d'étalonnage relient 3 types de variables :

- $M_k$  les mesures capteurs ( $k = 1..N_C$ ) ;
- $\xi$  les paramètres que l'on cherche à identifier ;
- des inconnues  $\Upsilon$  nécessaires pour modéliser les équations.

Ces inconnues  $\Upsilon = [\hat{\Upsilon}, \tilde{\Upsilon}_{1..N_C}]$  peuvent être :

- constantes  $\hat{\Upsilon}$  : leurs valeurs ne changent pas tout au long de la procédure d'étalonnage (par exemple des paramètres que l'on ne souhaite pas identifier) ;
- dépendantes de la pose  $\tilde{\Upsilon}_{k=1..N_C}$ .

Ainsi, le système d'équations obtenues avec les équations d'étalonnage reliant un jeu de mesures  $M = \{M_1, ..., M_{N_C}\}$  et les inconnues  $V = [\xi, \Upsilon]$  est :

$$f_{k,i}(M_k, V) = 0 + \varepsilon_{k,i}, \quad k = [1..N_C], i = [1..s] \quad (1.1)$$

Où  $\varepsilon$  est une erreur due aux incertitudes de mesures, aux approximations de modèle et à l'estimation des paramètres de ce modèle. Ce sont les erreurs  $\varepsilon_k$  que nous allons chercher à minimiser. Chaque mesure  $k$  fournit ici  $s$  équations qui modélisent le robot pour une pose donnée (pour le *MGD* on a  $s = n$  et pour le *MGI* on a  $s = m$ ).

La condition pour que ce système soit sur-contraint est donnée par :

$$N_C \times s > \dim(\xi) + \dim(\hat{Y}) + N_C \times \check{Y}_k \quad (1.2)$$

En présence d'incertitudes, un système sur-contraint n'a en général pas de solution  $\tilde{V}$  telle que

$$f_{k,i}(M_k, \tilde{V}) = 0, \forall k = [1 \dots N_C], \forall i = [1 \dots s]$$

Nous allons voir qu'il est possible de caractériser une solution aux sens des moindres carrés.

## 1.2.2 Méthodes pour l'identification de paramètres

L'approche classique pour définir la solution d'un problème d'identification (sur-contraint), comme (1.1), est la méthode des moindres carrés non-linéaires [Legendre 1825]. Elle est généralement obtenue avec un algorithme de type descente de gradient (Levenberg-Marquard, Newton-Raphson, etc), sans garantie sur la découverte du minimum global. Mais d'autres solutions peuvent être définies, et des méthodes de résolution nouvelles ont fait leur apparition.

### 1.2.2.1 Critères des moindres carrés

Une solution au système sur-contraint 1.1 peut être définie au sens des moindres carrés, sous l'hypothèse que l'espérance  $E(\varepsilon_k) = 0$ . C'est à dire si  $\varepsilon_k$  est un bruit blanc. Alors une solution aux moindres carrés  $\hat{\xi}$  minimise le critère  $F^T F$  avec  $F = [f_1(M_1, \hat{\xi}), \dots, f_{N_C}(M_{N_C}, \hat{\xi})]^T$ .

Si on se place dans le cas classique et linéaire (voir figure 1.14) où l'on cherche les paramètres ( $\alpha$  et  $\beta$ ) de telle sorte à faire passer une droite (le modèle  $y = \alpha x + \beta$ ) par des points (les mesures  $\{(x_1, y_1), \dots, (x_{N_C}, y_{N_C})\}$ ). Alors la fonction d'étalonnage est  $f_k((x_k, y_k), \alpha, \beta) = y_k - (\alpha x_k + \beta)$  et le critère  $F^T F$  est la somme des distances au carré.

Mais d'autres critères plus riches peuvent être utilisés.

**Moindres carrés avec poids :** La technique des moindres carrés avec poids (pondérés) permet d'introduire une notion de priorité sur les mesures [Wampler 1992]. Cette méthode utilise un critère pondéré ( $F^T \Sigma_F F$ ). La matrice de poids  $\Sigma_F$  permet de pondérer le critère basique  $F^T F$ . Une approximation linéaire de  $\Sigma_F$  peut être obtenue avec  $\Sigma_F = J_M^T \Sigma_M J_M$  où  $J_M = \frac{\partial F}{\partial M}$ . La matrice de poids  $\Sigma_M$  est obtenue en estimant la variance des mesures  $\sigma_{M_k}^2$  et, en supposant que les covariances sont nulles (pas de lien entre la mesure  $k$  et la mesure  $k+1$ ). La variance  $\sigma_{M_k}^2$  est obtenue à partir de l'écart type  $\sigma_{M_k}$ , qui n'est autre que la tolérance des appareils de mesure. La matrice de poids s'écrit donc  $\Sigma_M = \frac{1}{\sigma_{M_k}^2} Id_{N_C}$ .

**Régression des distances orthogonales :** La régression des distances orthogonales (ODR) prend en considération les possibles erreurs de mesure [Boggs 1987]. Cette méthode identifie, en plus des variables  $\xi$ , les mesures  $M_k$  qui permettront de diminuer le critère précédent ( $F^T \Sigma_F F$ ). Pour cela, elle utilise le critère  $(F^T \Sigma_F F) + (M^T \Sigma_M M)$ , où  $M$  est la différence entre les estimées initiales des mesures et leurs valeurs courantes  $M = [M_1 - \hat{M}_1, \dots, M_{N_C} - \hat{M}_{N_C}]^T$ . L'utilisation

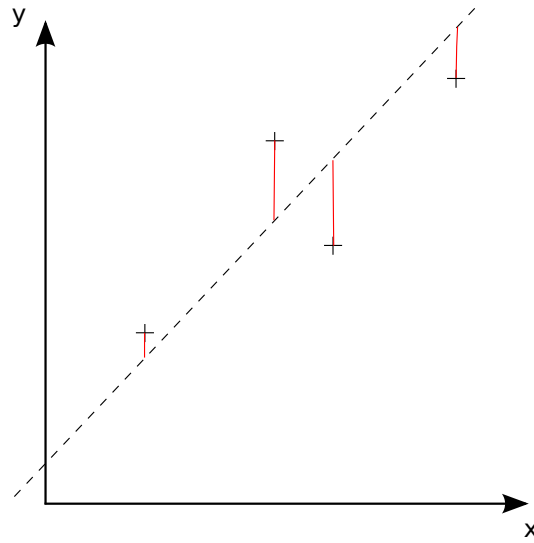


FIG. 1.14 – Exemple d'utilisation des moindres carrés basiques

de pondération sur les équations d'étalonnage  $\Sigma_F$  et sur les mesures  $\Sigma_M$  permet également d'ajouter une notion de poids. Les deux peuvent être calculées comme précédemment.

Si on se replace dans le cas précédent alors le critère supplémentaire  $M^T M$  correspond à l'écart entre la mesure et sa valeur courante lors de l'identification, dans ce cas  $M_k = x_k - \hat{x}_k$ . Le critère total est la somme des distances, en rouge sur la figure 1.15, au carré.

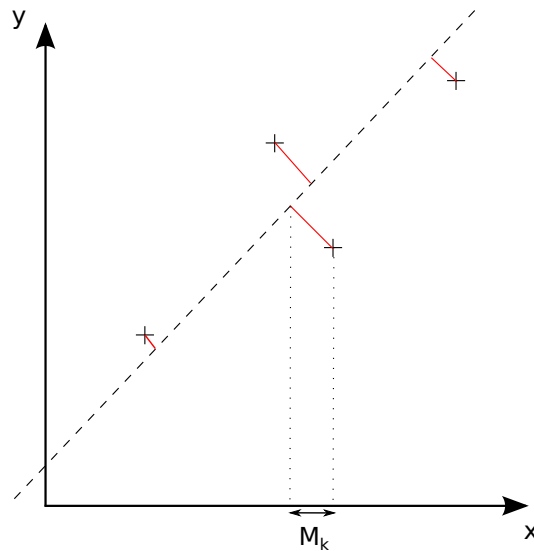


FIG. 1.15 – Exemple d'utilisation de la régression des distances orthogonales

**Contrôle de déviation :** Le critère  $\chi^2$  permet un contrôle de déviation de l'identification des inconnues. Il est utilisé par [Patel 2000]. Il est utile lors d'un étalonnage en condition difficile (bruit de mesure important). En posant  $V = [V_1 - \hat{V}_1, \dots, V_{N_C} - \hat{V}_{N_C}]$ , le critère considéré est  $(F^T \Sigma_F F) + (M^T \Sigma_M M) + (V^T \Sigma_V V)$ . Ici la pondération sur les inconnues  $\Sigma_V$  est nécessaire afin d'autoriser un éloignement des estimées initiales, tout en garantissant une non divergence. Cette matrice de poids sur les inconnues est obtenue avec la tolérance sur les paramètres, et en supposant qu'il n'y a pas de covariance entre les inconnues :  $J_V = \frac{1}{\sigma_{V_i}^2} Id_{N_C}$ .

Si on se place à nouveau dans le cas précédent alors le critère  $V^T V$  peut être vu comme un encadrement de  $\alpha$  et  $\beta$  (en vert sur la figure 1.16).

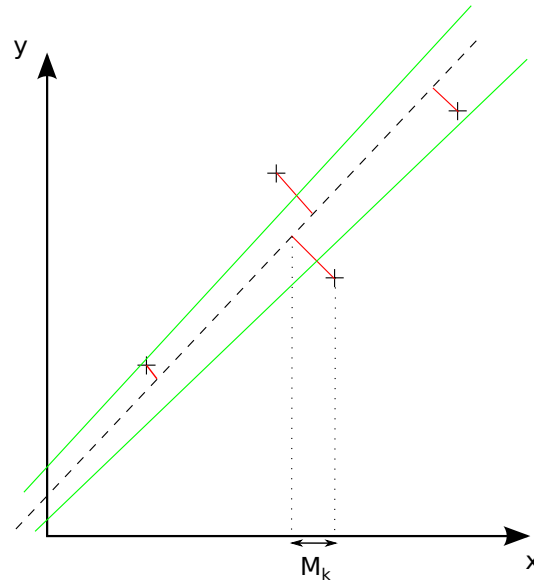


FIG. 1.16 – Exemple d'utilisation du contrôle de déviation.

### 1.2.2.2 Résolution des moindres carrés non linéaires

Une solution pour  $V$  aux équations d'étalonnage du système (1.1) peut être obtenue suivant différents critères. Une méthode de résolution classique des moindres carrés non linéaires est basée sur un schéma itératif utilisant un développement limité en série de Taylor au premier ordre, c'est la méthode de *Gauss-Newton*. On pourrait également utiliser la méthode de *Newton*, qui utilise un développement au second ordre, la méthode de *Levenberg-Marquardt* qui est une version adaptative de *Gauss-Newton*.

Le principe de la méthode de *Gauss-Newton* est de résoudre itérativement le système linéaire sur-contraint suivant :

$$J_V \cdot \Delta V = \Delta F \quad (1.3)$$

où la Jacobienne  $J_V = \begin{pmatrix} \frac{\partial f_1}{\partial V_1} & \cdots & \frac{\partial f_1}{\partial V_p} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_N}{\partial V_1} & \cdots & \frac{\partial f_N}{\partial V_p} \end{pmatrix}$

Le système (1.3) peut être résolu avec une décomposition en valeurs singulières (SVD) de  $J_V$ . Cette décomposition donne  $J_V = USW^T$ , et ainsi, grâce aux propriétés des matrices  $U, S, W$ , on obtient  $\Delta V = WS^+U^T\Delta F$  (où  $S^+$  est la pseudo-inverse de  $S$ , obtenue en remplaçant chaque terme non nul de sa diagonale par son inverse et en transposant la matrice résultat).

De plus, afin que le système admette une solution, une condition sur le rang de la jacobienne (égal au nombre de termes non nuls de la diagonale de  $S$ ) en fonction du nombre d'inconnues doit être vérifiée [Besnard 2001] :

$$\begin{aligned} \text{rang}(J_V) &= \dim(V) \\ &= \dim(\xi) + \dim(\hat{\Upsilon}) + N_C \times \tilde{\Upsilon}_k \end{aligned}$$

### 1.2.2.3 Les méthodes alternatives

L'amélioration de la précision d'un robot passe par la diminution des incertitudes de ses paramètres, de son modèle et de ses capteurs (voir section 1.2). Le problème de l'étalonnage consiste à modéliser ces incertitudes (à des degrés différents) dans le but d'identifier au mieux certains paramètres.

Si les moindres carrés prennent en considération les incertitudes sur les paramètres et sur les capteurs (voir section 1.2.2.1), d'autres méthodes, provenant pour la plupart de l'analyse de risque et des probabilités peuvent également permettre la modélisation des incertitudes d'un système [Montgomery 2009]. Toutes ne peuvent pas s'adapter à un problème d'étalonnage d'un robot (passage à l'échelle, non linéarité des équations d'étalonnage, identification des paramètres, nécessitant un système carré), mais certaines donnent des clés intéressantes.

**Le filtrage :** Le filtre le plus classique est probablement le filtre de Kalman. La méthode de Kalman est une procédure d'estimation dynamique de paramètres en fonction du temps [Gautier 2001]. C'est à dire que les valeurs des paramètres sont corrigées au fur et à mesure que de nouvelles observations sont effectuées. On peut le considérer comme une approche dynamique des moindres carrés.

**Les méthodes Bayésiennes :** Une introduction aux approches Bayésiennes est donnée dans [Lee 2004] et [Gelman 1995]<sup>3</sup>. Ces méthodes utilisent la notion d'*a priori*, en plus des informations apportées par les observations, pour tester les valeurs d'un paramètre. Ces *a priori*, sur le modèle ou les paramètres (leur distribution), peuvent être revus au fur et à mesure de la résolution à l'aide de poids. Ces méthodes sont utilisées plutôt en économie.

**Les p-box :** Une "probability box" [Ferson 2002], ou encadrement de probabilités, permet de définir des nombres incertains en encadrant leur valeur à l'aide de deux distributions. La manipulation de ces p-box est réalisée avec une analyse

<sup>3</sup>Certains prêtent à Alan Turing les premières définitions, retrouvées dans un travail personnel.

sur les distributions bornées issue de la théorie des probabilités [Chebyshev 1874, Markov 1886].

**L’approche aléatoire :** Le principe de la simulation consiste dans des tests répétés de valeurs pour les paramètres et la comparaison *a posteriori* du résultat. La simulation de Monte-Carlo [Ulam 1949] en est la version la plus évoluée. Une approche empirique inspirée d’un processus utilisé en métallurgie est le *recuit simulé*. Cette méthode est utilisée en robotique comme par exemple pour la recherche de poses d’étalonnage dans [Zhuang 1994].

**L’arithmétique floue :** L’arithmétique floue permet de propager une incertitude sur les paramètres ou les observations dans un modèle arithmétique [Kentel 2005]. Dans cette représentation, il est possible de définir des nombres avec des bornes et une distribution entre ces bornes. Certains résultats ont été obtenus en robotique avec la logique floue [Su 2011].

**L’analyse par intervalles :** Cette méthode utilise des intervalles pour représenter les incertitudes [Moore 1966]. En effet, une représentation par intervalles permet de considérer toutes les valeurs possibles d’une variable sur un domaine continu de variation. L’arithmétique associée permet de propager ces incertitudes mais aussi de prendre en compte les erreurs d’arrondi des ordinateurs. Les travaux réalisés ces dernières années en font un outil performant pour l’étalonnage des robots [Daney 2006].

### 1.2.3 Dans le cadre de l’étalonnage

Un problème d’étalonnage conduit à la résolution d’un système sur-contraint en présence d’incertitudes (voir section 1.2). Ces incertitudes peuvent provenir des mesures, d’une tolérance donnée sur un paramètre ou, plus rarement considérées de la modélisation (nous verrons un cas d’incertitude de modèle au Chapitre 3).

Plusieurs approches peuvent permettre de modéliser ces incertitudes, cependant la plupart utilisent un *a priori* sur la distribution de ces incertitudes (filtrage ou moindres carrés par exemple) ou engendrent un problème calculatoire (l’arithmétique floue par exemple).

Dans le cas beaucoup plus général, où ces incertitudes peuvent être bornées, les intervalles offrent une représentation satisfaisante et permettent des calculs complexes et certifiés.

## 1.3 Analyse par intervalles

La représentation des incertitudes au moyen d’intervalles a été utilisée en robotique pour résoudre le *MGD* des robots parallèles [Merlet 2004b]. Elle a également permis de définir l’espace de travail d’un robot parallèle à câbles [Gouttefarde 2006]. La conception d’un manipulateur a été réalisée avec l’analyse par intervalles [Chablat 2004]. Un étalonnage des robots parallèles a été proposé dans [Daney 2006]. L’analyse par intervalles permet donc de résoudre des problèmes complexes, avec certification des résultats, en prenant en compte les incertitudes, et cela sans *a priori* sur leur distribution [Moore 1966, Neumaier 1990, Jaulin 2001, Hansen 2003].

Une arithmétique par intervalles a tout d’abord été introduite par [Moore 1966].

### 1.3.1 Arithmétique par intervalles

Un intervalle  $[x_i] = [\underline{x}_i, \overline{x}_i]$  définit l'ensemble des réels  $x_i$  tels que  $\underline{x}_i \leq x_i \leq \overline{x}_i$ .  $\mathbb{IR}$  représente l'ensemble des intervalles sur les réels.

La taille ou le diamètre de  $[x_i]$  est  $w([x_i]) = \overline{x}_i - \underline{x}_i$ . Son rayon est noté  $rad([x_i]) = (\overline{x}_i - \underline{x}_i)/2$  et son milieu est noté  $\text{Mid}([x_i])$ .

Une **boîte**  $[x]$  est un vecteur d'intervalles tel que  $[x] = \begin{pmatrix} [x_1] \\ \vdots \\ [x_i] \\ \vdots \\ [x_n] \end{pmatrix}$ .

Le diamètre de  $[x]$  est défini par  $\max_i w([x_i])$ .

L'*arithmétique par intervalles* [Moore 1966] étend dans  $\mathbb{IR}$  les fonctions élémentaires de  $\mathbb{R}$ . Elle fournit donc les opérateurs de base suivants :

- La somme +
- La soustraction −
- Le produit \*
- La division /

Et les fonctions mathématiques usuelles :

- De trigonométrie  $\sin x, \cos x, \dots$
- La fonction carrée  $x^2$
- La racine carrée  $\sqrt{x}$
- $\log(x), e^x, \dots$

Par exemple, la somme au sens intervalle :  $[x_1] + [x_2] = [\underline{x}_1 + \underline{x}_2, \overline{x}_1 + \overline{x}_2]$ , contient l'image de la fonction somme sur tous ses arguments. Cet encadrement définit ce que l'on nomme une *extension par intervalles*.

#### Définition 2 (L'extension d'une fonction à $\mathbb{IR}$ )

Considérant une fonction  $f : \mathbb{R}^n \rightarrow \mathbb{R}$ .

$[f] : \mathbb{IR}^n \rightarrow \mathbb{IR}$  est une **extension** de  $f$  aux intervalles si :

$$\begin{aligned} \forall [x] \in \mathbb{IR}^n \quad & [f]([x]) \supseteq \{f(x), x \in [x]\} \\ \forall x \in \mathbb{R}^n \quad & f(x) = [f](x) \end{aligned}$$

Dans un contexte classique, l'expression d'une fonction  $f$  est toujours une composition de fonctions élémentaires ( $\sin x, \sqrt{x}, x^2$  etc.). L'**extension naturelle**  $[f]_N$  est alors simplement la composition des opérateurs d'intervalles correspondants. L'**extension de Taylor**, notée  $[f]_T$ , utilise le développement en séries de Taylor au premier ou second ordre de la fonction et calcule leur extension naturelle. Son expression est donnée par  $[f]_T([x]) = f(\hat{x}) + [J_f]([x]).([x] - \hat{x})$ , où  $\hat{x} \in [x]$  et  $J_f = \frac{\partial f}{\partial x}$  la dérivée au premier ordre de  $f$ .

**Exemple :** Considérant  $f(x_1, x_2) = 3x_1^2 + x_2^2 + x_1 * x_2$  dans la boîte  $[x] = \begin{pmatrix} [-1, 3] \\ [-1, 5] \end{pmatrix}$ .

L'extension naturelle est :  $[f]_N([x_1], [x_2]) = 3 * [-1, 3]^2 + [-1, 5]^2 + [-1, 3] * [-1, 5] = [0, 27] + [0, 25] + [-5, 15] = [-5, 67]$ . Cette extension utilise la version intervalle de  $x_1^2 = [0, 9]$ , car en développant et en utilisant le produit naturel on trouve  $x_1 * x_1 = [-3, 9]$ .



Les dérivées de  $f$  se définissent alors :

$$J_f = \begin{pmatrix} \frac{\partial f}{\partial x_1}(x_1, x_2) \\ \frac{\partial f}{\partial x_2}(x_1, x_2) \end{pmatrix} = \begin{pmatrix} 6x_1 + x_2 \\ x_1 + 2x_2 \end{pmatrix} \text{ et } [J_f]_N\left(\begin{pmatrix} [-1, 3] \\ [-1, 5] \end{pmatrix}\right) = \begin{pmatrix} [-7, 23] \\ [-3, 13] \end{pmatrix}$$

L'extension de Taylor au premier ordre avec  $\hat{x} = \text{Mid}([x]) = \begin{pmatrix} 1 \\ 2 \end{pmatrix}$  donne :

$$[f]_T([x_1], [x_2]) = f(\hat{x}) + [J_f]_N \times ([x] - \hat{x}) = 9 + \begin{pmatrix} [-7, 23] \\ [-3, 13] \end{pmatrix} \times \begin{pmatrix} [-2, 2] \\ [-3, 3] \end{pmatrix} = [-76, 94].$$

On a vu que suivant l'extension utilisée, le résultat n'était pas le même. Ce phénomène est du à la décorrélation des variables provoquée par le développement des équations et aux occurrences multiples des variables. Il entraîne une surestimation de l'intervalle résultat.

### Définition 3 (Surestimation d'un ensemble)

Considérant l'ensemble  $F = \{f(x), x \in [x]\}$ , l'extension par intervalle  $[f]([x])$  est une surestimation de  $F$  notée :  $\square F = [f]([x])$ .

### 1.3.2 Schéma de Newton par intervalles

Dans  $\mathbb{R}$ , l'approche classique pour la résolution d'un système d'équations non linéaires consiste à itérativement l'approximer par un système linéaire (son développement de Taylor au premier ordre par exemple) puis à résoudre le système linéarisé obtenu (avec le pivot de Gauss par exemple). Cette méthode générique, le schéma de Newton, est traduite aux intervalles.

En reprenant les notations de l'extension de Taylor (dont l'équation initiale est la même), on peut écrire une relation adaptée aux intervalles pour les systèmes non linéaires.

$$[J_f([x])]s = -f(\hat{x}) \quad (1.4)$$

où  $[J_f([x])]$  est l'extension naturelle de la jacobienne du système d'équations  $f$  et  $s$  la solution au problème linéaire  $[A]s = [b]$  avec  $[A] = J_f([x])$  et  $[b] = -f(\hat{x})$ . Ainsi, il a été montré [Hansen 2003] que les solutions du système étaient incluses, de manière certifiée, dans  $[x] \cap s + \hat{x}$ .

Pour le choix de  $\hat{x} \in [x]$ , on prend généralement le milieu de  $[x]$ . Une étape importante de ce schéma est la résolution du système linéaire d'intervalles obtenu par la linéarisation.

### 1.3.3 Systèmes linéaires d'intervalles

Les systèmes linéaires d'intervalles [Rump 1987, Neumaier 1990, Hansen 2003] s'écrivent sous la forme :

$$[A]x = [b] \quad (1.5)$$

Cette représentation permet de définir ces systèmes comme un ensemble de systèmes linéaires dont les valeurs de  $A$  et de  $b$  sont prises dans des intervalles.

La résolution consiste à trouver, sous la forme d'un vecteur d'intervalles (une boîte), par exemple un sur-ensemble  $\square \Sigma_{\exists, \exists}$  de

$$\Sigma_{\exists, \exists}([A], [b]) = \{x, \exists A \in [A], \exists b \in [b], Ax = b\} \quad (1.6)$$

L'allure  $\Sigma_{\exists,\exists}([A], [b])$  est bien connue en dimension 2 (voir figure 1.17), on l'appelle fréquemment le papillon (dû à sa forme évocatrice). Cette forme est due à la variation de la pente  $A \in [A]$  et de l'ordonnée à l'origine  $b \in [b]$ .

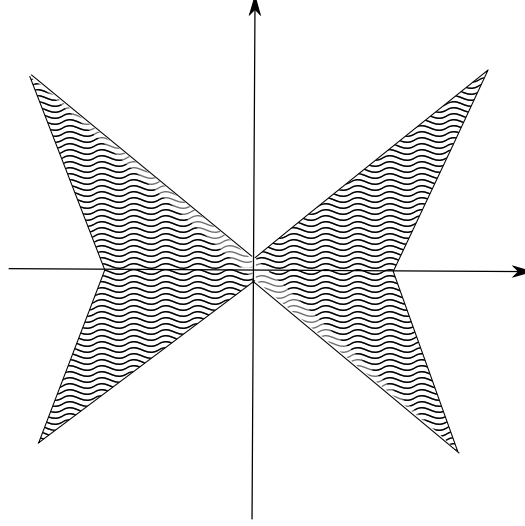


FIG. 1.17 – Exemple de solution  $\Sigma_{\exists,\exists}([A], [b])$

Il est important de noter que l'ensemble  $\Sigma_{\exists,\exists}([A], [b])$  n'est généralement pas convexe et le calcul du vecteur  $[x] \supset \Sigma_{\exists,\exists}([A], [b])$  optimal (dans le sens où la boîte englobante est de diamètre minimal) est un problème NP-dur [Rohn 1994, Kreinovich 1997].

Quatre différents ensembles peuvent être définis au moyen des quantificateurs :

- $\Sigma_{\exists,\exists}([A], [b]) = \{x, \exists A \in [A], \exists b \in [b], Ax = b\}$
- $\Sigma_{\exists,\forall}([A], [b]) = \{x, \exists A \in [A], \forall b \in [b], Ax = b\}$
- $\Sigma_{\forall,\forall}([A], [b]) = \{x, \forall A \in [A], \forall b \in [b], Ax = b\}$
- $\Sigma_{\forall,\exists}([A], [b]) = \{x, \forall A \in [A], \exists b \in [b], Ax = b\}$

### 1.3.3.1 Interprétation

Shary [Shary 2002] donne une interprétation de ces ensembles dans le cas d'un problème inverse linéaire. Supposons que la sortie  $y$  d'un système physique soit donnée par  $y = Ax$ , avec  $A$  une matrice de transfert et  $x$  l'entrée du système.

Alors il définit :

- si  $x \in [x] \subset \Sigma_{\exists,\exists}([A], [b])$ , il y aura une matrice de transfert  $A \in [A]$  telle que une sortie du système  $y$  sera dans  $[b]$ . C'est l'ensemble des solutions **simples**.
- si  $x \in [x] \subset \Sigma_{\forall,\exists}([A], [b])$ , la sortie du système  $y$  sera toujours dans  $[b]$  quelle que soit la matrice de transfert  $A \in [A]$ . C'est l'ensemble des solutions **tolérables**.
- si  $x \in [x] \subset \Sigma_{\exists,\forall}([A], [b])$ , il y aura une matrice de transfert  $A \in [A]$  telle que toutes les sorties du système  $y$  seront dans  $[b]$ . C'est l'ensemble des solutions **contrôlables**.
- si  $x \in [x] \subset \Sigma_{\forall,\forall}([A], [b])$ , toute les fonctions de transfert de  $[A]$  engendreront toutes les sorties de  $[b]$ . Cet ensemble est généralement vide mais pourrait

avoir un sens dans certains cas.

On parle alors de A-incertitude (pour  $\forall$ ) et de E-incertitude (pour  $\exists$ ).

### 1.3.4 Caractérisation de la solution

Avec la représentation des intervalles, il est possible de définir au moins quatre types de réponses à la résolution d'un problème (voir figure 1.18). Pour un ensemble  $\Sigma_x$  que l'on cherche à caractériser, on peut définir :

- L'enveloppe extérieure  $[x]_{out} \supset \Sigma_x$  : elle contient **tous** les éléments de  $\Sigma_x$ , mais le surestime ;
- Une boîte intérieure  $[x]_{in} \subset \Sigma_x$  : elle ne contient que des éléments de  $\Sigma_x$ , mais pas tous ;
- Un pavage  $\mathcal{L}_x$  : une liste de boîtes intérieures ;
- Un élément intérieur de  $\Sigma_x$  tel qu'il optimise une fonction de coût.

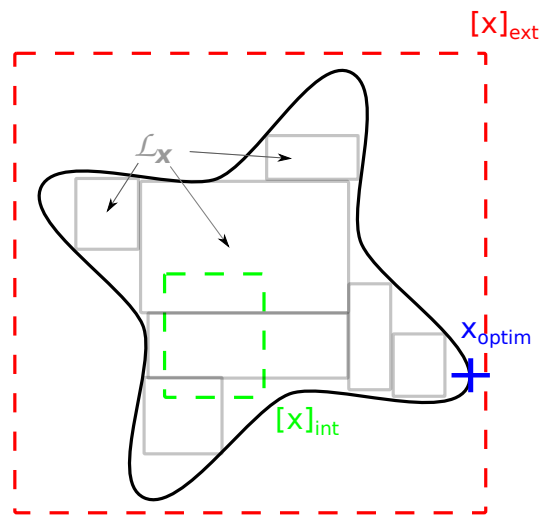


FIG. 1.18 – Enveloppe extérieure  $[x]_{out}$ , boîte intérieure  $[x]_{in}$ , pavage  $\mathcal{L}_x$  et élément intérieur  $x_{optim}$  optimisant le coût “le plus à droite”.

L'enveloppe extérieure et la boîte intérieure peuvent répondre à notre problème d'étalonnage (Chapitre 2 et 5). Le pavage définit un sous espace de travail (Chapitre 3) et l'optimisation de coût permet l'étude d'un modèle de câble (Chapitre 3).

### 1.3.5 Méthodes de traitement des systèmes d'intervalles

Plusieurs réponses peuvent être données pour caractériser un ensemble de solutions : une enveloppe extérieure, une boîte intérieure, un pavage ou un élément solution optimisant un coût. Afin d'obtenir ces différentes caractérisations, diverses méthodes peuvent être utilisées, issues de l'algèbre linéaire, de l'analyse ou de la programmation par contraintes. Le traitement des systèmes de contraintes consiste à encadrer le plus finement possible la ou les solutions de ces systèmes, ou à prouver qu'il n'y a aucune solution (l'ensemble est donc vide).

### 1.3.5.1 Systèmes linéaires d'intervalles bien contraints

Une majorité des travaux sur la résolution des systèmes linéaires d'intervalles se concentrent sur les systèmes bien contraints (où la matrice  $[A]$  est carrée).

Le calcul d'une boîte englobant l'ensemble  $\Sigma_{\exists,\exists}([A], [b])$ , notée  $\Box_{\Sigma_{\exists,\exists}}([A], [b])$ , nécessite un algorithme spécifique afin de se rapprocher d'une solution optimale. Deux familles de méthodes existent (et parfois donnent naissance à des méthodes hybrides) : les méthodes directes et les méthodes de point fixe.

**Méthodes directes :** Par opposition aux méthodes de point fixe, les méthodes directes ne contractent pas vers une solution englobante, mais permettent d'obtenir directement  $[x] \supset \Sigma_{\exists,\exists}([A], [b])$  [Aberth 1997, Oettli 1965, Rohn 1994]. Parmi ces méthodes, on peut citer :

- *Gauss-élimination* : une version intervalle de l'algorithme basique d'élimination du pivot de Gauss est une méthode directe qui donne de bons résultats [Neumaier 1990]. Elle permet de triangulariser le problème et d'obtenir une sur-estimation correcte de l'ensemble solution.
- *Quadrant* : un algorithme proposé par [Oettli 1965, Aberth 1997] s'appuie sur le théorème de Oettli et Pragger [Oettli 1964] pour résoudre le système quadrant par quadrant, ce qui convexifie le problème du papillon. Le théorème donne comme résultat :

$$x \in \Sigma_{\exists,\exists}([A], [b]) \Leftrightarrow |A_c x - b_c| \leq \Delta b + \Delta A |x|$$

Le problème est alors de composer avec les valeurs absolues. Une des méthodes est la résolution cadran par cadran.

- *Hansen-Bliek-Rohn-Ning-Kearfott* : sous certaines conditions, cette méthode permet de calculer directement chaque élément  $[x_i]$  de  $[x]$  indépendamment.

**Méthodes itératives contractantes :** Les méthodes itératives s'appuient sur le théorème du point fixe de Brouwer (Lemma 1) afin de garantir, sous certaines conditions, une contraction vers  $\Box_{\Sigma_{\exists,\exists}}$  [Rump 1987, Hansen 2003].

**Lemme 1** *Toute fonction continue  $f$  d'un espace convexe et compacte  $\mathcal{K}$  vers lui-même possède un point fixe tel que  $f(x_0) = x_0$ .*

Les méthodes itératives contractantes reposent sur l'écriture d'un contracteur, appelé itérativement jusqu'à atteindre ce point fixe.

Pour la résolution des systèmes linéaires d'intervalles, elles emploient des contracteurs issus de l'algèbre, en voici quelques uns :

- *Contracteur de Gauss-Seidel* : il utilise un découpage de la matrice  $[A]$  en une matrice diagonale et une matrice dont la diagonale est nulle.
- *Contracteur de Krawczyk linéaire* : il est basé sur une réécriture de  $Ax = b$  en  $x = b + (I - A)x$ .

**Préconditionnement :** Les méthodes de résolution de système linéaire peuvent être très sensibles au conditionnement de  $[A]$ . Ainsi il peut être intéressant de preconditionner le système en le réécrivant sous la forme :

$$(C[A])x = C[b] \tag{1.7}$$

La multiplication à gauche de  $[A]$  et  $[b]$  par la même matrice non nulle peut permettre une meilleure résolution [Hansen 2003, Kearfott 1990, Shary 2002].

Le préconditionneur le plus utilisé est  $C = A_c^{-1}$ . Il implique que  $A_c$  soit inversible. Il est également possible de choisir une matrice de préconditionnement telle que  $CC^T = A_c$  (factorisation de cholesky) afin de préserver la symétrie éventuelle de  $A_c$ .

On peut citer les autres factorisation  $QR$ ,  $LU$  et  $SVD$ . Par exemple avec une factorisation  $QR$ , le problème se réécrit :  $Rx = Q^T b$ , ou  $R^T Rx = A^T b$  (équation semi-normale).

On peut également utiliser les décompositions de  $A_c$  avec une matrice diagonale (Gauss-Seidel), une matrice symétrique ou encore la matrice identité.

Les seules “règles” (basées sur des observations [Guenette 2011]) sont :

- $C$  se choisit entre deux cas extrémaux  $C = A_c^{-1}$  et  $C = I$  ;
- $C$  doit être proche de  $A_c^{-1}$ , c’est à dire  $\|CA_c - I\| < \varepsilon$  ;
- il faut respecter au mieux les propriétés de  $A_c$  (symétrie par exemple) ;
- $CA_c$  doit être presque normale ;
- les valeurs propres de  $CA_c$  doivent être regroupées.

### 1.3.5.2 Deux méthodes en détails

Parmi les méthodes qui permettent d’obtenir une réponse aux caractérisations définies précédemment, deux ont retenu notre attention et sont présentées en détails.

**Hansen-Bliik-Rohn-Ning-Kearfott :** Sous certaines conditions sur  $[A]$ , la méthode de Hansen-Bliik-Rohn-Ning-Kearfott [Neumaier 2000] permet d’obtenir  $[x] \supset \Sigma_{\exists, \exists}([A], [b])$ .

Posons  $u = \langle [A] \rangle^{-1} |[b]|$ , avec  $\langle [A] \rangle$  la matrice dite de comparaison [Hansen 2003] et définie telle que :

$$\langle [A] \rangle_{ii} = \min\{|\alpha|, \alpha \in [A]_{ii}\}$$

et pour  $k \neq i$  :

$$\langle [A] \rangle_{ik} = \max\{|\alpha|, \alpha \in [A]_{ik}\}$$

Posons  $d_i = (\langle [A] \rangle^{-1})_{ii}$  et

$$\alpha_i = \langle [A] \rangle_{ii} - 1/d_i, \quad \beta_i = u_i/d_i - |[b]_i|$$

Alors on peut montrer que  $\square \Sigma_{\exists, \exists} \subset [x]$  défini par :

$$[x]_i = \frac{[b]_i + [-\beta_i, \beta_i]}{[A]_{ii} + [-\alpha_i, \alpha_i]} \quad (1.8)$$

Les conditions sur la matrice  $[A]$  sont qu’elles doit être régulière et diagonale dominante. En pratique, cela impose généralement l’utilisation d’un préconditionnement et la solution  $[x]$  obtenue est alors l’optimale (l’enveloppe extérieure la moins sur-évaluée possible).

**Méthode pour calculer un vecteur inclus dans  $\Sigma_{\forall,\exists}([A], [b])$  :** Dans sa thèse Beaumont [Beaumont 1999] propose deux algorithmes permettant de calculer un vecteur d'intervalles inclus (on parle généralement de boîte intérieure) dans  $\Sigma_{\forall,\exists}([A], [b])$ . Ils sont basés sur les *Quadrants* de [Oettli 1965, Aberth 1997]. Le plus robuste des deux [Beaumont 1999] est donné en 1.3.5.2.

Il repose sur le théorème de [Rohn 1985] suivant :

**Théorème 1.3.1**  $x \in \Sigma_{\forall,\exists}([A], [b]) \Leftrightarrow \exists (x_1, x_2), x = x_1 - x_2$  où  $(x_1, x_2)$  est la solution du système d'inégalités linéaires suivant :

$$\begin{pmatrix} B_1 \\ -I_{2n} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \leq \begin{pmatrix} b' \\ 0 \end{pmatrix},$$

où

$$B_1 = \begin{pmatrix} A_c + \Delta A & -(A_c - \Delta A) \\ -(A_c - \Delta A) & A_c + \Delta A \end{pmatrix} \text{ et } b' = \begin{pmatrix} \bar{b} \\ -\underline{b} \end{pmatrix}$$

Ce théorème donne une description d'un point de  $\Sigma_{\forall,\exists}([A], [b])$ . L'algorithme 1.3.5.2 utilise ensuite un second théorème qui permet de décrire un sous ensemble de  $\Sigma_{\forall,\exists}([A], [b])$ .

**Théorème 1.3.2** Posons  $x_0^+ = \frac{|x_0| + x_0}{2}$  et  $x_0^- = \frac{|x_0| - x_0}{2}$ .

Soit  $x_0 \in \Sigma_{\forall,\exists}([A], [b])$ ,  $b_1 = b' - B_1 \begin{pmatrix} x_0^+ \\ x_0^- \end{pmatrix}$ ,

et

$$S_1 = \left\{ z, z = \begin{pmatrix} x_0 - 2y_2 \\ x_0 + 2y_1 \end{pmatrix}, \begin{pmatrix} B_1 + |B_1| \\ -I_{2n} \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} \leq \begin{pmatrix} b_1 \\ 0 \end{pmatrix} \right\}$$

$$\text{Alors, } z = \begin{pmatrix} z_1 \\ z_2 \end{pmatrix} \in S_1 \Rightarrow [z_1, z_2] \subset \Sigma_{\forall,\exists}([A], [b]).$$

Les preuves de ces deux théorèmes sont disponibles dans [Beaumont 1999]. Un critère est ajouté afin de définir l'optimalité d'un vecteur d'intervalles de  $\Sigma_{\forall,\exists}([A], [b])$ . Il consiste à exprimer un coût linéaire en fonction de chaque variable  $x_i$  inversement proportionnel à son diamètre ( $\bar{x}_i - \underline{x}_i$ ). La minimisation de ce critère peut être réalisée avec un algorithme d'optimisation linéaire sous contraintes. Cet algorithme nécessite de résoudre un programme linéaire de  $5n$  contraintes et de  $2n$  variables.

### 1.3.6 Contracteurs

Les algorithmes de filtrage ou dits de contraction sont très utiles dans les méthodes de résolution par intervalles de systèmes d'équations. Ils permettent de réduire l'espace de recherche, une boîte, afin d'encadrer un ensemble solution le plus finement possible. Dans certains cas, ils peuvent permettre de conclure rapidement à l'absence de solutions dans une boîte donnée.

**Algorithme 1** Algorithme 8 de Beaumont

---

Fixer  $w_{min}$  un critère sur le diamètre de  $[x]$   
 Résoudre  $A_c x_0 = b_c$   
 $x_0^+ = \frac{|x_0| + x_0}{2}$   
 $x_0^- = \frac{|x_0| - x_0}{2}$   
 $B_1 := \begin{pmatrix} A_c + \Delta A & -(A_c - \Delta A) \\ -(A_c - \Delta A) & A_c + \Delta A \end{pmatrix}$   
 $b_1 := \begin{pmatrix} \bar{b} \\ -\underline{b} \end{pmatrix} - B_1 \begin{pmatrix} x_0^+ \\ x_0^- \end{pmatrix}$   
 $C_1 := \begin{pmatrix} B_1 + |B_1| & \\ -I_{2n} & \\ -2I_n & -2I_n \end{pmatrix}$   
 $c_1 := \begin{pmatrix} b_1 \\ 0_{2n} \\ -w_{min} \end{pmatrix}$   
 Trouver  $y_1$  et  $y_2$  tels qu'ils minimisent  $\{\sum_{i=1}^n c_i(y_1(i) + y_2(i)) \text{ avec } C_1 \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} \leq c_1\}$   
 $[x] = [x_0 - 2y_2, x_0 + 2y_1]$

---

**1.3.6.1 Opérateur de Newton intervalles**

L'opérateur de Newton est une adaptation aux intervalles des algorithmes de l'analyse numérique. Il permet la contraction d'une boîte par la résolution récursive d'un système linéarisé en utilisant la jacobienne du système (voir section 1.3.2).

**Algorithme 2** Schéma de Newton

---

**répéter**  
 $[A] = J([x])$   
 $[b] = F(\text{Mid}[x])$   
 Résoudre  $[A]s = [b]$  avec une méthode de résolution de système linéaire  
 $[x] = [x] \cap s + \text{Mid}[x]$   
**jusqu'à** Atteinte d'un point fixe

---

Cette linéarisation nécessite la résolution de ce système linéaire. En changeant les méthodes utilisées pour cela, Newton intervalles s'est enrichi d'un grand ensemble de variantes. Toutes les méthodes présentées (directes ou de point fixe) peuvent être utilisées pour cette résolution.

**1.3.6.2 Contracteurs issus de la résolution des systèmes linaires**

Les méthodes de résolution des systèmes linéaires présentées à la section 1.3.5.1 permettent de résoudre l'étape la plus importante du schéma de Newton. En effet, plus cette résolution sera efficace, plus le schéma contractera la boîte solution  $[x]$ . Ces méthodes de résolution (Hansen-blik, Rump, etc) sont donc vus dans ce cas comme des contracteurs.

### 1.3.6.3 Contracteurs issus de la programmation par contraintes

Des algorithmes innovants issus de la programmation par contraintes peuvent être utilisés. Ils consistent à considérer les contraintes une par une, les contracter en exploitant leur structure (monotonie, occurrence des variables, etc.) et à propager la contraction sur les autres contraintes du système considéré.

**Algorithmes de propagation de contraintes :** Ces algorithmes de contractions reposent sur l'écriture d'un algorithme de point fixe incrémental. Le principe est de contracter les intervalles des variables correspondantes en considérant une seule contrainte à la fois, puis de propager la contraction éventuelle d'une variable sur les autres (ou une partie des autres) contraintes.

Les algorithmes de propagation diffèrent principalement entre eux par le contracteur utilisé sur la contrainte prise individuellement, comme par exemple les contracteurs : **HC4-Revise** et **Mohc-Revise**.

**HC4-Revise :** Le contracteur **HC4-Revise** [Benhamou 1999] contracte une boîte par rapport à une seule contrainte. Il utilise une représentation arborescente qu'il parcourt dans les deux sens. La contraction est ainsi rapide et efficace. L'algorithme de propagation utilisant le contracteur **HC4-Revise** s'appelle **HC4**.

**Mohc-Revise :** Le contracteur **Mohc-Revise** [Araya 2010] contracte une boîte par rapport à une seule contrainte en combinant le contracteur **HC4-Revise** à une procédure exploitant l'éventuelle monotonie de la contrainte par rapport à une variable. En effet, la contraction est optimale lorsque la contrainte est monotone sur ces variables (au flottant près). L'algorithme de propagation exploitant le contracteur **Mohc-Revise** se nomme **Mohc**.

**3BCID :** Un outil améliorant la contraction est l'algorithme de contraction **3BCID** [Trombettoni 2007], une variante de la **3B** [Lhomme 1993].

La **3B** est un algorithme de rognage qui consiste en 3 étapes :

- Découper en tranches l'intervalle correspondant à chaque variable ;
- Contracter chaque tranche se trouvant aux extrémités de l'intervalle (avec **Mohc** ou **HC4** par exemple) ;
- Écarter les tranches si aucune solution ne s'y trouve.

Ainsi en répétant cela sur toutes les tranches aux bornes, l'intervalle initial est contracté.

L'algorithme **3BCID** ajoute 2 étapes :

- Contracter les tranches restantes (celles qui ne peuvent être éliminées) ;
- Réaliser l'union de ces tranches.

**q-intersection :** La **q-intersection** [Jaulin 2009a] permet le calcul d'une intersection relaxée. Ce contracteur retourne la plus petite boîte résultat de l'intersection de  $q$  boîtes parmi un plus grand nombre. Ce contracteur est décrit et utilisé au chapitre 5.



#### 1.3.6.4 Algorithmes de branchement

Les méthodes par intervalles peuvent approximer avec précision un ensemble de solutions d'un système de contraintes  $F(X) = 0$  au travers d'un ensemble de boîtes. Elles permettent ainsi de trouver plusieurs solutions disjointes à un problème d'intervalles au moyen d'algorithmes de branchement. En ajoutant une fonction objectif à un système de contraintes, une méthode de branchement peut trouver la solution optimisant cette fonction.

**Schéma de “Branch & Prune” :** Le processus simple de résolution des algorithmes de branchement commence par une boîte initiale  $[x]$  représentant l'espace de recherche et construit un arbre de recherche, suivant un schéma de *Branch & Prune* :

- *Prune* : on calcule l'image de  $[x]$  par  $F$  (méthode de l'évaluation intervalle). Si  $0 \notin F([x])$ , alors  $[x]$  ne contient pas de solution au système et cette boîte est écartée.
- *Branch* : dans le cas contraire, la boîte  $[x]$  est coupée en deux, on parle alors de bisection, et on relance la recherche récursivement sur la boîte gauche et la boîte droite.
- La récursion stoppe lorsque les boîtes restantes ont atteint une précision donnée.

L'algorithme du “Branch & Prune” est naïf et souvent peu efficace. Mais une variante permet d'améliorer et d'accélérer ce schéma par l'ajout de contracteurs. Ces algorithmes permettent en effet d'écarter rapidement des boîtes de taille importante et également de réduire aux bornes les intervalles de recherche sans perte de solution. Cet ajout permet d'accélérer le processus de résolution.

**Schéma “Branch & Contract” :** Le processus de résolution commence par une boîte initiale  $[x]$  représentant l'espace de recherche et construit un arbre de recherche, suivant un schéma de *Branch & Contract* :

- *Contract* : un algorithme de filtrage (aussi appelé *contracteur*) réduit les bornes de la boîte  $[x]$  sans perte de solution.
- *Branch* : dans le cas contraire, la boîte  $[x]$  est bisectée, et on relance la recherche récursivement sur la boîte gauche et la boîte droite.
- La récursion stoppe lorsque les boîtes restantes ont atteint une précision donnée.

Les algorithmes de contraction comprennent des algorithmes de type Newton issus de la communauté de l'analyse numérique [Moore 1966] et des algorithmes provenant de la programmation par contraintes. Tous ceux présentés à la section 1.3.6 peuvent ainsi être utilisés. Une variante de cette méthode permet l'optimisation sous contraintes.

**Schéma “Branch & Bound” :** Les méthodes par intervalles peuvent également traiter un problème plus complexe, l'optimisation sous contraintes. L'optimisation sous contraintes consiste à trouver une solution qui minimise une fonction de *coût* et qui satisfait un ensemble de contraintes. Pour cela, la stratégie suit un schéma de *Branch & Bound* [Horst 1996].

- *Contract* : un algorithme de filtrage (aussi appelé *contracteur*) réduit les bornes de la boîte  $[x]$  sans perte de solution.

- *Bound* : les bornes de la fonction de coût sont mises à jour suivant le principe décrit ci après.
- *Branch* : l'algorithme sélectionne une boîte  $[x]$  dans l'arbre de recherche. Il choisit de manière heuristique une variable  $x_i \in x$  et bissecte  $[x_i]$ . On relance la recherche récursivement sur la boîte gauche et la boîte droite.
- La récursion stoppe lorsque les boîtes restantes ont atteint une précision donnée.

*Bound* : La borne inférieure est mise à jour en trouvant un point de coût moins bon que tous les autres points de la boîte étudiée (généralement non faisable, c'est à dire ne satisfaisant pas toutes les contraintes). Il est alors inutile de vérifier les boîtes ayant un coût inférieur car elles ne peuvent être réalisables (leur coût serait inférieure au minimum global). L'ensemble solution est généralement approximé par une technique de linéarisation et un algorithme d'optimisation linéaire trouve le meilleur point dans le polytope sur-estimé. On note  $lb$  (de l'anglais "lower bound") la valeur minimale des bornes inférieures de toutes les boîtes traitées par la stratégie d'optimisation.

De même,  $ub$  (de l'anglais "upper bound") est le coût d'un point de la boîte courante satisfaisant les contraintes et minimisant au mieux la fonction objective. Cette phase de mise à jour de la borne supérieure est réalisée avec des techniques de recherche locale ou avec des méthodes plus sophistiquées [Trombettoni 2011].

L'algorithme se termine lorsque  $ub - lb$  atteint la précision souhaitée.

**Heuristiques de bisection :** L'ordre dans lequel les variables sont sélectionnées pour la bisection est crucial. L'heuristique de la *Smear* [Kearfott 1990] ou une variante [Trombettoni 2011] consiste à bissecter plus fréquemment une variable dont son intervalle de recherche est large et si les dérivées partielles des contraintes par rapport à cette variable sont significatives. Il existe de nombreuses autres heuristiques pour choisir la variable à bissecter [Granvilliers 2012].

### 1.3.6.5 Programmation par contracteurs

Une approche récente dans les techniques basées sur les intervalles est le "Contractor Programming", ou programmation par contracteurs [Chabert 2009]. Dans cette approche, une généralisation des contracteurs est proposée. Elle consiste à voir comme contracteur toute méthode prenant en entrée une boîte et qui, au moyen d'une ou de plusieurs contraintes ou de tout type d'informations, rend la boîte réduite.

Par exemple, l'intersection peut être vue comme un contracteur dans le sens où on a toujours  $w([x_1 \cap x_2]) \leq w([x_1])$ . Pour simplifier encore, toute procédure qui réduit le diamètre d'une boîte est un contracteur (à condition de respecter certaines propriétés détaillées dans [Chabert 2009]). Cette vision uniformisée nous a permis de définir la **q-intersection** comme un contracteur.

De plus, un contracteur accepte en entrée une boîte qui peut elle même être issue d'un ou de plusieurs contracteurs, ainsi ce formalisme unifié permet la composition de contracteurs. Un étalonnage reposant sur une composition de contracteurs est défini et exploité au chapitre 5.

## 1.4 Conclusion

Les robots parallèles à câbles ont de nombreux avantages mais engendrent également quelques incertitudes supplémentaires. La modélisation des câbles est complexe et nécessite d'être étudiée afin d'améliorer la précision du manipulateur CoGiRo.

Afin d'améliorer la précision d'un robot, l'étalonnage de ses paramètres géométriques est une étape nécessaire. Cette étape se découpe en plusieurs phases : la modélisation du robot, la recherche de poses pour l'étalonnage, la prise des mesures, l'étude de l'identifiabilité, l'identification et la validation. L'identification des paramètres, en quelque sorte le cœur de l'étalonnage nécessite l'utilisation de méthodes performantes. Une de ces méthodes, permettant en plus une certification du résultat de l'étalonnage, est de représenter toutes les incertitudes du robot au moyen des intervalles.

L'identification des paramètres du robot revient avec cette représentation à caractériser autrement la solution d'un étalonnage et de résoudre un système non linéaire sur-contraint par intervalles. Pour cela, des méthodes basées sur l'algèbre et sur la programmation par contraintes peuvent être utilisées.

Dans le chapitre suivant, nous allons adapter certaines de ces méthodes à l'identification des paramètres d'un système mécanique simple.

# Résolution des systèmes non linéaires sur-contraints d'intervalles, étalonnage d'un mécanisme simple

---

## Sommaire

<b>2.1</b>	<b>Schéma de Newton sur-contraint</b>	<b>42</b>
<b>2.2</b>	<b>Caractérisation de la solution du système linéarisé sur-contraint</b>	<b>42</b>
2.2.1	Approche des moindres carrés	43
2.2.2	Considération du reste	44
2.2.3	Caractérisation de la solution	45
<b>2.3</b>	<b>Algorithmes et nouvelle définition du résultat de l'étalonnage</b>	<b>47</b>
2.3.1	Calcul d'une boîte extérieure	47
2.3.2	Calcul d'une boîte intérieure	47
2.3.3	Résultat de l'étalonnage	48
<b>2.4</b>	<b>Étalonnage d'un mécanisme simple</b>	<b>48</b>
2.4.1	Présentation du mécanisme	48
2.4.2	Identification	48
2.4.3	Mesures, incertitudes et estimée initiale	49
<b>2.5</b>	<b>Approches classiques</b>	<b>51</b>
2.5.1	Barycentre des sous systèmes carrés	51
2.5.2	Système complet et la jacobienne	52
2.5.3	Moindres carrés	52
<b>2.6</b>	<b>Approche des intervalles</b>	<b>52</b>
2.6.1	Données et leur incertitude	52
2.6.2	Identification des paramètres avec les intervalles	52
<b>2.7</b>	<b>Discussion et conclusion</b>	<b>56</b>

---

L'étalonnage d'un robot, ou d'un mécanisme en général, revient à identifier les paramètres de son modèle à partir de mesures de son état. Dans la pratique, les mesures sont bruitées. Afin de minimiser l'effet de ce bruit sur le résultat de l'étalonnage, il est indispensable de multiplier les mesures de l'état du robot afin de moyenner les erreurs de mesures. L'étalonnage implique donc couramment de fournir une solution à un système sur-contraint lors de la phase de l'identification des paramètres (voir section 1.2). Une solution telle que l'ensemble des équations s'annulent n'existe pas en présence de bruit de mesure. Afin de fournir une solution à un problème d'identification de paramètres, une approche des moindres carrés est

généralement utilisée. Elle consiste à définir une solution “au mieux”, c’est à dire minimisant la somme des carrés des résidus des équations (voir section 1.2.2.1).

L’approche des moindres carrés nécessite un *a priori* sur la distribution des incertitudes de mesures (bruit gaussien ou distribution normale), qu’il est, en pratique, difficile de vérifier. De plus, aucune garantie n’est obtenue sur le résultat de l’identification. Mais il existe d’autres façons de définir une solution à ces systèmes sur-contraints.

Les intervalles, avec une approche ensembliste, permettent de modéliser simplement les incertitudes, à la condition unique que l’on puisse les borner. Cette approche permet de proposer des descriptions de la solution d’un étalonnage, ainsi qu’une certification du résultat obtenu.

Si certaines méthodes comme la programmation par contraintes permettent un traitement compatible aux problèmes non linéaires sur-contraints d’intervalles (nous le verrons au chapitre 5), les méthodes globales, c’est à dire qui considèrent toutes les contraintes simultanément, nécessitent d’être adaptées aux cas sur-contraints. Dans ce chapitre, nous allons voir que le schéma de Newton intervalles (présenté à la section 1.3.2) peut être adapté au traitement d’un système non linéaire sur-contraint. L’approche choisie pour la résolution du système linéarisé peut également fournir différentes caractérisations pour le résultat d’un étalonnage.

## 2.1 Schéma de Newton sur-contraint

Le schéma de Newton intervalles (présenté à la section 1.3.2) permet de résoudre les systèmes non linéaires. Il consiste à itérativement approximer le système non linéaire par son développement de Taylor au premier degré, puis à résoudre ce système linéaire obtenu.

Notons  $N([x], x_0) = x_0 - \Sigma([J([x])], [F(x_0)])$ , où  $x_0 \in [x]$  (généralement le milieu) et  $\Sigma([A], [b]) = \{s : As = b\}$  l’ensemble des solutions du système linéaire d’intervalles où  $[A] = [J([x])]$  et  $[b] = [F(x_0)]$ . Alors une itération du schéma de Newton pour résoudre  $F(x) = 0$  s’écrit :

$$[x] \leftarrow [x] \cap N([x], x_0) \quad (2.1)$$

La solution donnée par le schéma de Newton,  $[x]$ , est alors caractérisée par la définition de l’ensemble  $\Sigma([A], [b])$ . Dans le cas de l’étalonnage, on a vu que le système linéarisé était sur-contraint ( $[A]$  est une matrice d’intervalles  $m \times n$  et  $[b]$  est un vecteur d’intervalles  $m$ , avec  $m > n$ ). Les caractérisations des solutions qu’il est possible de trouver pour ce système vont donc dépendre de la manière de traiter un système sur-contraint.

## 2.2 Caractérisation de la solution du système linéarisé sur-contraint

Nous avons vu les caractérisations possibles pour la solution d’un système linéaire d’intervalles à la section 1.3.3. Il est possible de définir plusieurs réponses à un problème d’intervalles comme, par exemple, une boîte intérieure ou extérieure et également d’utiliser des quantificateurs.

## 2.2. Caractérisation de la solution du système linéarisé sur-contraint 43

Ces caractérisations peuvent être utilisées pour les solutions des systèmes sur-contraints. Cependant, les approches permettant de définir une solution pour un système sur-contraint ont également un impact sur la caractérisation du résultat. Par exemple, dans  $\mathbb{R}$ , les moindres carrés définissent la solution comme étant celle minimisant le carré des résidus.

Il est possible de caractériser la solution au système linéaire sur-contraint par  $\Sigma_{\exists, \exists}([A]_{m,n}, [b]_m)$  ( $m > n$ ). Cette caractérisation a été utilisée dans [Daney 2006]. Cependant, une majorité des travaux sur la résolution des systèmes linéaires d'intervalles se concentrent sur les systèmes bien contraints (où la matrice  $[A]$  est carrée). Il peut donc être intéressant de réécrire notre problème en un problème bien contraint. Pour cela, nous allons voir deux méthodes : les moindres carrés intervalles et les équations de Bjorck.

### 2.2.1 Approche des moindres carrés

Pour définir une solution à un système linéaire sur-contraint d'intervalles  $[A]_{m,n}s_n = [b]_m$  avec  $m > n$ , l'approche aux moindres carrés peut être adaptée aux intervalles.

Ainsi [Bentbib 2002] propose :

$$\Sigma([A], [b]) = \{s | \exists A \in [A], \exists b \in [b] : \|As - b\|_2 = \min_{\forall \hat{s} \in [s]} \|A\hat{s} - b\|_2\} \quad (2.2)$$

En algèbre linéaire, la solution utilisant une pseudo inverse est une solution au sens des moindres carrés. Elle revient à déterminer  $s$  tel que  $(A^T A)s = A^T b$ .

Cette approche est couramment utilisée dans le domaine des intervalles [Deif 1988, Gay 1988, Popova 2005].

[Neumaier 1990] propose une inversion de matrice au sens des moindres carrés notées  $[A]^L$  telle que :  $[A]^L[b] = \square\{A^+b | A \in [A], b \in [b]\}$ . On peut noter que  $[A]^H[b] \subseteq [A]^L[b]$ , avec  $H$  la Hull-inverse, c'est à dire une surestimation de l'ensemble des matrices inverses de  $[A]$ . Elle réduit normalement la surévaluation due à la pseudo-inverse intervalle.

La caractérisation de la solution au système linéaire sur-contraint au sens des moindres carrés est définie par :

$$\Sigma_{\exists, \exists}([A], [b]) = \{s | \exists A \in [A], \exists b \in [b] : A^T A s = A^T b\} \quad (2.3)$$

**Discussion :** Dans l'idée des moindres carrés avec poids, [Schön 2005] a ajouté aux moindres carrés par intervalles la notion de poids (la matrice  $P$ ) et définit :

$$\Sigma_{\exists, \exists}([A], [b]) = \{s | \exists A \in [A], \exists b \in [b] : A^T P A s = A^T P b\} \quad (2.4)$$

En gardant cette approche des moindres carrés, [Gay 1988, Deif 1988] ajoutent une notion de statistique et [Kreinovich 1997] celle des probabilités avec les p-box. Quant à [Schön 2005], il utilise la représentation des zonotopes en remplacement des boîtes.

Dans l'équation (2.2), la norme 2 pourrait être remplacée par une autre norme. [Jaulin 2001] a étudié pour cela la norme  $\infty$ , cette norme entraîne un problème de type minimax [Chebyshev 1874]. [Manteuffel 1981] a étudié pour sa part les différentes normes sur les matrices intervalles et a conclu que la norme 2 était la plus robuste.

### 2.2.2 Considération du reste

Dans le domaine des réels, il est possible de montrer qu'une solution aux sens des moindres carrés implique une notion de reste.

Soit  $\hat{s}$ , une solution au système linéaire  $A^T A s = A^T b$ , c'est à dire la solution aux moindres carrés du système sur-contraint  $As = b$ . Posons  $r = b - A\hat{s}$ , le reste, et en multipliant par  $A^T$  à gauche on obtient :  $A^T r = A^T b - A^T A \hat{s}$ . Mais par construction on a  $A^T A \hat{s} - A^T b = 0$ , et on en conclut  $A^T r = 0$ .

De cette manipulation on retrouve les équations de Björck [Björck 1967, Higham 2002] :

$$As + r = b, \quad A^T r = 0 \quad (2.5)$$

Ces équations peuvent être combinées en un problème linéaire de la forme :

$$\begin{bmatrix} I & A \\ A^T & 0 \end{bmatrix} \begin{bmatrix} r \\ s \end{bmatrix} = \begin{bmatrix} b \\ 0 \end{bmatrix} \quad (2.6)$$

Ce résultat est plus intéressant qu'une approche aux moindres carrés pour transformer un système sur-contraint en un système carré, tout en évitant de faire absorber par la solution  $s$  les incertitudes de  $A$  et de  $b$  [Higham 2002]. Ces incertitudes sont en effet principalement définies par le reste  $r$ , qui est lui même dans le noyau de  $A^T$ .

Comme avec la méthode de la pseudo inverse, cette méthode a été transposée aux systèmes sur-contraints d'intervalles [Rump 1987, Deif 1988, Popova 2005] :

$$\begin{bmatrix} I & [A] \\ [A]^T & 0 \end{bmatrix} \begin{bmatrix} r \\ s \end{bmatrix} = \begin{bmatrix} [b] \\ 0 \end{bmatrix} \quad (2.7)$$

Avec l'utilisation d'une matrice d'intervalles, les équations de Björck entraîne une surestimation (voir section 1.3.1) moindre qu'avec une pseudo-inverse car ne nécessite aucun produit de matrice.

[Hansen 1967] propose une réécriture de ce système sous forme d'inégalités, sous la condition que  $s \geq 0$  :

$$\begin{aligned} r + \underline{A}s &\leq \bar{b} & A_1^T r &\leq 0 \\ r + \bar{A}s &\geq \underline{b} & A_2^T r &\geq 0 \end{aligned} \quad (2.8)$$

où  $A_1$  et  $A_2$  sont les bornes supérieure et inférieure de  $[A]$  excepté pour les valeurs correspondantes aux  $r$  négatifs.

Nous retiendrons la caractérisation de la solution au système linéaire sur-contraint définie par :

$$\Sigma_{\exists, \exists}([A], [b]) = \{s : \exists A \in [A], \exists b \in [b], \exists r, \begin{bmatrix} I & A \\ A^T & 0 \end{bmatrix} \begin{bmatrix} r \\ s \end{bmatrix} = \begin{bmatrix} b \\ 0 \end{bmatrix}\} \quad (2.9)$$

#### 2.2.2.1 Propriétés de la solution :

Une solution obtenue  $(\hat{r}, \hat{s}) \in \Sigma_{\exists, \exists}$  avec cette méthode implique donc :

$$\exists A \in [A], \exists b \in [b] : A\hat{s} + \hat{r} = b, \quad A^T \hat{r} = 0$$

## 2.2. Caractérisation de la solution du système linéarisé sur-contraint 45

En multipliant la première relation par  $A^T$ , on retrouve la définition de la solution au sens des moindres carrés :

$$A^T A \hat{s} = A^T b$$

Ainsi, une boîte englobante  $\square\Sigma_{\exists,\exists}$  (obtenue avec une méthode de la section 1.3.5.1) est une sur-estimation de l'ensemble des solutions aux moindres carrés du système sur-contraint  $([A], [b])$ .

Nous pouvons distinguer trois cas lors de la résolution d'un système non linéaire sur-contraint d'intervalles :

- il y a une solution qui annule toutes les contraintes, elle est donc incluse dans  $\square\Sigma_{\exists,\exists}$  ;
- il n'y a pas de solution qui annule toutes les contraintes :
  - la solution aux moindres carrés est dans l'estimée initiale, elle est donc incluse dans  $\square\Sigma_{\exists,\exists}$  ;
  - la solution aux moindres carrés n'est pas dans l'estimée initiale,  $\Sigma_{\exists,\exists} = \emptyset$  (ceci est dû au schéma de Newton qui nécessite une estimée initiale englobant la solution)

Nous allons visualiser les deux premiers cas sur un exemple classique : déterminer l'intersection de trois cercles dont le rayon varie dans un intervalle donné.

Dans l'image du haut de la figure 2.1, les trois cercles s'intersectent et  $\square\Sigma_{\exists,\exists}$  est bien une sur-estimation de l'intersection. Il est important de noter qu'alors  $0 \in [\hat{r}]$ . Dans l'image du bas de la figure 2.1, les trois cercles ne s'intersectent pas (le centre d'un cercle est décalé, mais on aurait aussi pu diminuer l'intervalle définissant le rayon) et  $\square\Sigma_{\exists,\exists}$  est bien une sur-estimation de l'ensemble des solutions aux moindres carrés. Il est important de noter qu'alors  $0 \notin [\hat{r}]$  (aucun terme  $\hat{s} \in \square\Sigma_{\exists,\exists}$  n'annulera toutes les contraintes).

Nous avons vu qu'il était possible que les cercles ne s'intersectent pas. Ceci peut arriver lors de l'étalonnage et peut provenir d'une mesure aberrante.

**Mesures aberrantes au sens intervalle :** Il s'agit d'une mesure (en général, mais on pourrait considérer un paramètre aberrant) dont l'incertitude est mal estimée (trop restreinte) ou dont l'estimation est complètement erronée. Par exemple, elle peut provenir d'un appareil de mesure mal réglé dont la tolérance est plus grande qu'indiquée par le constructeur. Elle peut aussi être issue d'un capteur en défaut dont la mesure n'a aucun sens. Et enfin, une erreur humaine ne peut être entièrement écartée (recopie manuelle des mesures, erreur de saisie, etc).

Les intervalles permettent de définir une mesure aberrante :

### Définition 4 (Mesure aberrante)

*$\hat{x}$  est une mesure aberrante d'une grandeur, si la valeur réelle  $x$  de cette grandeur est telle que :  $x \notin \hat{x} + [\sigma_x]$ , où  $[\sigma_x]$  est la tolérance donnée au moyen de mesure.*

### 2.2.3 Caractérisation de la solution

Une fois l'approche choisie pour la définition de la solution, telle que la somme des carrés des résidus soit minimisée (pseudo-inverse) ou telle que les résidus soient confinés dans le noyau du système (équations de Bjorck), il est possible de caractériser plusieurs solutions grâce aux quantificateurs ( $\exists$  ou  $\forall$ ) ou suivant le type de solution proposée (boîte intérieure ou extérieure, pavage ou optimum).



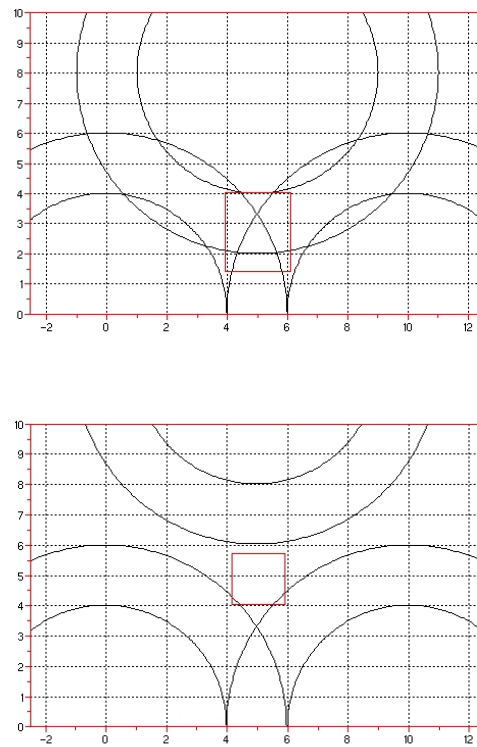


FIG. 2.1 – Solutions  $\square_{\Sigma, \exists}$  obtenues avec les équations de Bjorck à une intersection de cercles épais : en haut lorsqu'ils s'intersectent, en bas lorsqu'ils ne s'intersectent pas.

Dans le cadre de l'identification de paramètres et l'utilisation du résultat de l'étalonnage pour la commande du robot, nous considérons comme intéressant la possibilité d'avoir :

- Une enveloppe extérieure de la solution afin de borner au plus près les paramètres du robot en fonction des mesures effectuées. Cette démarche revient à diminuer l'incertitude sur les paramètres.
- Une boîte intérieure dans laquelle un jeu de paramètres peut être pris et utilisé dans la commande du robot, tout en garantissant que ces paramètres satisferont l'ensemble des mesures effectuées.

## 2.3 Algorithmes et nouvelle définition du résultat de l'étalonnage

Dans le schéma de Newton, l'étape de résolution du système linéarisé sur-contraint peut être réalisé au moyen de différentes approches, comme par exemple la pseudo-inverse ou le reste (voir section 2.2), afin de définir différentes solutions, comme une enveloppe extérieure ou une boîte intérieure de Beaumont (voir section 1.3.5.2).

Ceci nous fournit 4 différents algorithmes.

### 2.3.1 Calcul d'une boîte extérieure

L'étape de résolution du système linéarisé sur-contraint (le calcul de  $[s]$ ) pour définir une boîte extérieure peut être réalisée de deux façons :

- $[s]_{out} = \square \Sigma_{\exists, \exists} \{s | \exists A \in [A], \exists b \in [b] : A^T A s = A^T b\}$
- $[s]_{out} = \square \Sigma_{\exists, \exists} \{s | \exists A \in [A], \exists b \in [b] : \begin{bmatrix} I & A \\ A^T & 0 \end{bmatrix} \begin{bmatrix} r \\ s \end{bmatrix} = \begin{bmatrix} b \\ 0 \end{bmatrix}\}$

Le calcul de  $[s]$  sera réalisé avec une méthode de Hansen-Blik (voir section 1.3.5.2).

La solution du schéma de Newton  $[x]_{out}$  aura la propriété qu'aucun point en dehors ne pourra être solution.

### 2.3.2 Calcul d'une boîte intérieure

L'étape de résolution du système linéarisé sur-contraint (le calcul de  $[s]$ ) afin de définir une boîte intérieure de Beaumont peut être réalisée de deux façons :

- $[s]_{in} \subset \Sigma_{\forall, \exists} \{s | \forall A \in [A], \exists b \in [b] : A^T A s = A^T b\}$
- $[s]_{in} \subset \Sigma_{\forall, \exists} \{s | \forall A \in [A], \exists b \in [b] : \begin{bmatrix} I & A \\ A^T & 0 \end{bmatrix} \begin{bmatrix} r \\ s \end{bmatrix} = \begin{bmatrix} b \\ 0 \end{bmatrix}\}$

Le calcul de  $[s]$  sera réalisé avec la méthode de Beaumont (voir section 1.3.5.2).

Le calcul d'une boîte intérieure de  $\Sigma_{\forall, \exists}$  dans le cadre d'un étalonnage et d'un schéma de Newton implique des propriétés fortes. La solution trouvée valide **tous** les intervalles de mesures qui ont permis d'instancier  $[A]$ . C'est à dire que la solution couvre toutes les mesures possibles effectuées ainsi que leur incertitude. De plus, le fait que la solution finale soit un sous ensemble de l'ensemble solution implique la propriété des boîtes intérieures, c'est à dire que n'importe quel point de  $[x]_{in}$  est solution de l'étalonnage.

La boîte intérieure de  $\Sigma_{\forall, \exists} \{s | \forall A \in [A], \exists b \in [b] : A^T A s = A^T b\}$  n'est pas une boîte intérieure de  $\Sigma_{\forall, \exists} \{s | \forall A \in [A], \exists b \in [b] : A s = b\}$  car la pseudo-inverse implique une sur-évaluation à cause de la multiplication par  $[A]^T$ .

Ces quatre algorithmes vont être appliqués à l'étalonnage d'un mécanisme simple dans la suite de ce chapitre en vue de les comparer.

### 2.3.3 Résultat de l'étalonnage

On a désormais les outils pour calculer une sur-estimation de l'ensemble solution aux équations d'étalonnage

$$[x]_{out} \supset \square \Sigma_{\exists, \exists}$$

et pour calculer un ensemble intérieure

$$[x]_{in} \subset \Sigma_{\forall, \exists}$$

On propose de répondre à un problème d'identification de paramètres en fournissant un résultat de la forme  $\{[x]_{in}; [x]_{out}\}$ .

Ainsi, on certifie que les paramètres ne peuvent prendre aucune valeur en dehors de  $[x]_{out}$  aux vues des mesures faites, et on garantie que  $[x]_{in}$  contient des paramètres satisfaisant toutes les mesures, c'est à dire dans l'ensemble tolérable pour le système [Shary 2002, Beaumont 1999].

Ce résultat est plus riche en informations, de plus certifiées, que ce qui pourrait être obtenu avec une approche des moindres carrés.

## 2.4 Étalonnage d'un mécanisme simple

On se propose d'étudier le cas d'un mécanisme simple afin de valider les résultats de l'identification de paramètres obtenus par les méthodes basées sur l'utilisation d'intervalles et présentées précédemment.

### 2.4.1 Présentation du mécanisme

Nous étudions un système plan constitué d'une barre de longueur  $L$  pivotant librement autour d'un pivot en  $(X_0, Y_0)$ . La position de l'extrémité de la barre est donnée par  $(x, y)$  (voir figure 2.2).

Les équations géométriques peuvent être écrites avec une équation de distance :  $(x - X_0)^2 + (y - Y_0)^2 = L^2$ .

Pour la simulation,  $L$  est considéré constant avec  $L = 5$ , et le pivot a pour coordonnées idéales  $(X_0, Y_0) = (-1, -1)$ .

### 2.4.2 Identification

Dans le système présenté, nous considérons  $(X_0, Y_0)$  comme étant les paramètres du système à identifier et  $(x, y)$  les sorties du système que l'on va mesurer. Ainsi chaque configuration  $i$  du système va nous donner une équation implicite du type :

$$f_i = (x_i - X_0)^2 + (y_i - Y_0)^2 - L^2 = 0 \quad (2.10)$$

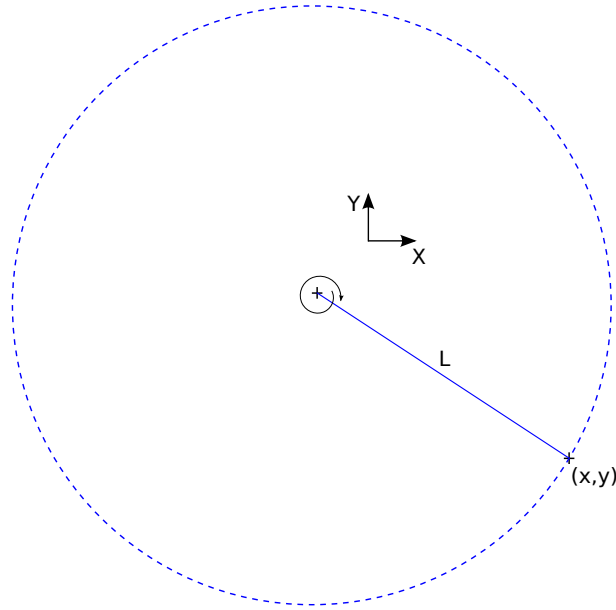


FIG. 2.2 – Schéma du mécanisme de la barre.

Par conséquent, si le système est placé dans  $N = 2$  configurations (différentes), il est défini par le système de deux équations quadratiques à deux inconnues  $(X_0, Y_0)$  :

$$\begin{cases} (x_1 - X_0)^2 + (y_1 - Y_0)^2 - L^2 = 0 & (1) \\ (x_2 - X_0)^2 + (y_2 - Y_0)^2 - L^2 = 0 & (2) \end{cases}$$

Géométriquement, cela revient à chercher l'intersection de deux cercles de rayon  $L$  et centrés en  $(x_i, y_i)$  (voir figure 2.3).

Ainsi, à chaque ajout d'une mesure, un cercle vient s'ajouter et l'intersection des cercles obtenus peut ne plus exister en présence d'incertitudes sur la mesure des couples  $(x_i, y_i)$  (voir figure 2.3).

### 2.4.3 Mesures, incertitudes et estimée initiale

La mesure des  $(x, y)$  fournit à chaque fois un couple  $(x_i, y_i)$ , ainsi on dispose de différents jeux de mesures. Nous avons fixé le nombre de mesures à 4, soit le double du nombre de paramètres à identifier.

Un jeu  $S_1$  de 4 mesures prises au hasard :

- $x_1=2.01$  et  $y_1=3.01$  ;
- $x_2=3.01$  et  $y_2=2.01$  ;
- $x_3=3.99$  et  $y_3=-1.02$  ;
- $x_4=-1.01$  et  $y_4=-5.98$  ;

Un jeu  $S_2$  de 4 mesures choisies afin d'améliorer le résultat de l'identification (les barres des 4 mesures sont orthogonales) :

- $x_1=-1.01$  et  $y_1=4.01$  ;
- $x_2=-5.99$  et  $y_2=-1.01$  ;
- $x_3=3.99$  et  $y_3=-1.02$  ;

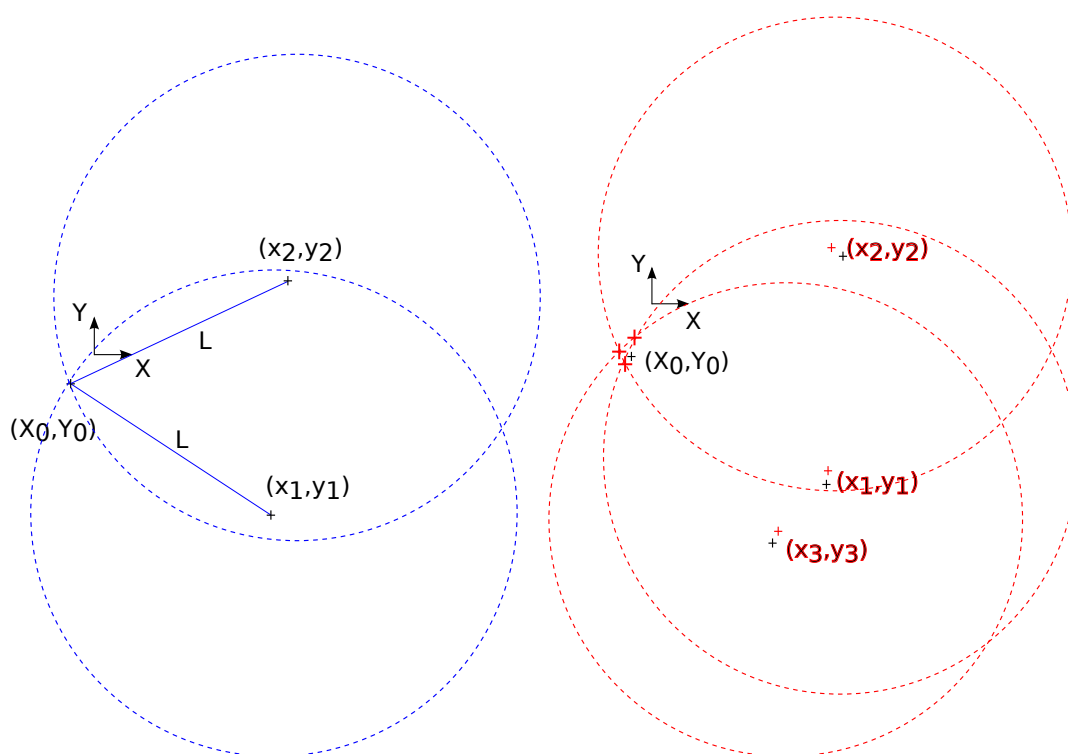


FIG. 2.3 – Le mécanisme placé dans deux et trois configurations de mesure.

–  $x_4 = -1.01$  et  $y_4 = -5.98$  ;

Et enfin un jeu  $S_3$  issu de  $S_2$  mais dont on remplace la quatrième mesure par une mesure aberrante (hors tolérance) :

–  $x_4 = 2.01$  et  $y_4 = 1.56$  ;

L'estimée initiale du couple de paramètres est obtenue en ajoutant un bruit aléatoire aux valeurs idéales. Il est fixé pour la suite de cette étude à  $(X_0, Y_0) = (-1.02, -0.95)$ .

## 2.5 Approches classiques

Les paramètres  $(X_0, Y_0)$  peuvent être identifiés avec des approches classiques du type barycentre ou moindres carrés. Afin de juger des résultats obtenus avec des méthodes basées sur l'utilisation des intervalles, ces différentes méthodes classiques ont été appliquées à notre problème didactique.

### 2.5.1 Barycentre des sous systèmes carrés

Avec plus de deux mesures comportant de l'incertitude, les cercles s'intersectent en plusieurs points distincts (voir figure 2.3). Avec nos jeux de quatre mesures, il est possible de calculer 6 intersections séparément et ensuite de calculer le barycentre de ces six points. Pour cela il est nécessaire de “piocher” deux équations parmi les quatre fournis par les mesures et de résoudre ce système (avec le résultant, des méthodes itératives de type Newton, des méthodes d'optimisation). Si la méthode fournit deux solutions à chaque système de deux équations, il est nécessaire de choisir celle qui est la plus proche de l'estimée initiale. Nous utilisons la méthode de Levenberg-Marquardt.

En utilisant les mesures du jeu  $S_1$ , on trouve les intersections :

- $(X_0, Y_0)_{12} = (-0.99, -0.99)$  ;
- $(X_0, Y_0)_{13} = (-1.0098, -0.975)$  ;
- $(X_0, Y_0)_{14} = (-1.0032, -0.98)$  ;
- $(X_0, Y_0)_{23} = (-1.0096, -0.9635)$  ;
- $(X_0, Y_0)_{24} = (-0.9974, -0.9800)$  ;
- $(X_0, Y_0)_{34} = (-1.0098, -0.98)$ .

Et le barycentre (moyenne) de ces 6 intersections est :

$$(X_0, Y_0) = (-1.003, -0.978)$$

En utilisant les mesures du jeu  $S_2$ , on trouve le barycentre :

$$(X_0, Y_0) = (-1.039, -0.937)$$

En utilisant les mesures du jeu  $S_3$ , on trouve le barycentre :

$$(X_0, Y_0) = (-1.219, -1.398)$$

Cette méthode combinatoire devient rapidement lourde à utiliser. De plus, si un des résultats de la résolution d'un système n'est pas bon (en présence d'un système mal conditionné ou d'une mesure aberrante), le barycentre peut en être fortement impacté.

### 2.5.2 Système complet et la jacobienne

Le système de quatre équations obtenu avec quatre mesures est :

$$F(X_0, Y_0) = \begin{cases} (x_1 - X_0)^2 + (y_1 - Y_0)^2 - L^2 = 0 & (1) \\ (x_2 - X_0)^2 + (y_2 - Y_0)^2 - L^2 = 0 & (2) \\ (x_3 - X_0)^2 + (y_3 - Y_0)^2 - L^2 = 0 & (3) \\ (x_4 - X_0)^2 + (y_4 - Y_0)^2 - L^2 = 0 & (4) \end{cases}$$

La jacobienne de ce système s'écrit alors :

$$J(X_0, Y_0) = 2 \cdot \begin{pmatrix} x_1 - X_0 & y_1 - Y_0 \\ x_2 - X_0 & y_2 - Y_0 \\ x_3 - X_0 & y_3 - Y_0 \\ x_4 - X_0 & y_4 - Y_0 \end{pmatrix}$$

### 2.5.3 Moindres carrés

Une approche aux moindres carrés consiste à trouver les paramètres  $(X_0, Y_0)$  tels qu'ils minimisent le critère  $F^T F$ .

Un algorithme d'optimisation est utilisé pour cette minimisation et les résultats sont :

- avec le jeu  $S_1 : (X_0, Y_0) = (-1.006, -0.977)$  ;
- avec le jeu  $S_2 : (X_0, Y_0) = (-0.999, -0.985)$  ;
- avec le jeu  $S_3 : (X_0, Y_0) = (-1.120, -1.321)$ .

## 2.6 Approche des intervalles

Les méthodes par intervalles présentées au début de ce chapitre ont été utilisées pour résoudre ce problème.

### 2.6.1 Données et leur incertitude

#### 2.6.1.1 Mesures

Il est possible de considérer l'incertitude (la tolérance) autour des mesures :  $[x_i] = x_i + [-0.04, 0.04]$  et  $[y_i] = y_i + [-0.06, 0.06]$ . Les anneaux obtenus (l'ensemble des cercles dont le centre est dans une boîte) des mesures des trois jeux sont illustrés à la figure 2.4.

#### 2.6.1.2 Estimée initiale

L'estimée initiale par intervalles est définie par la tolérance de fabrication comme  $[X_0] = [-1.12, -0.92]$  et  $[Y_0] = [-1.05, -0.85]$ .

### 2.6.2 Identification des paramètres avec les intervalles

#### 2.6.2.1 Boîtes extérieures $\square \Sigma_{\exists, \exists}$

Nous avons vu deux approches pour calculer l'enveloppe extérieure des paramètres lors d'un étalonnage. Avant de les appliquer à l'étalonnage de ce mécanisme

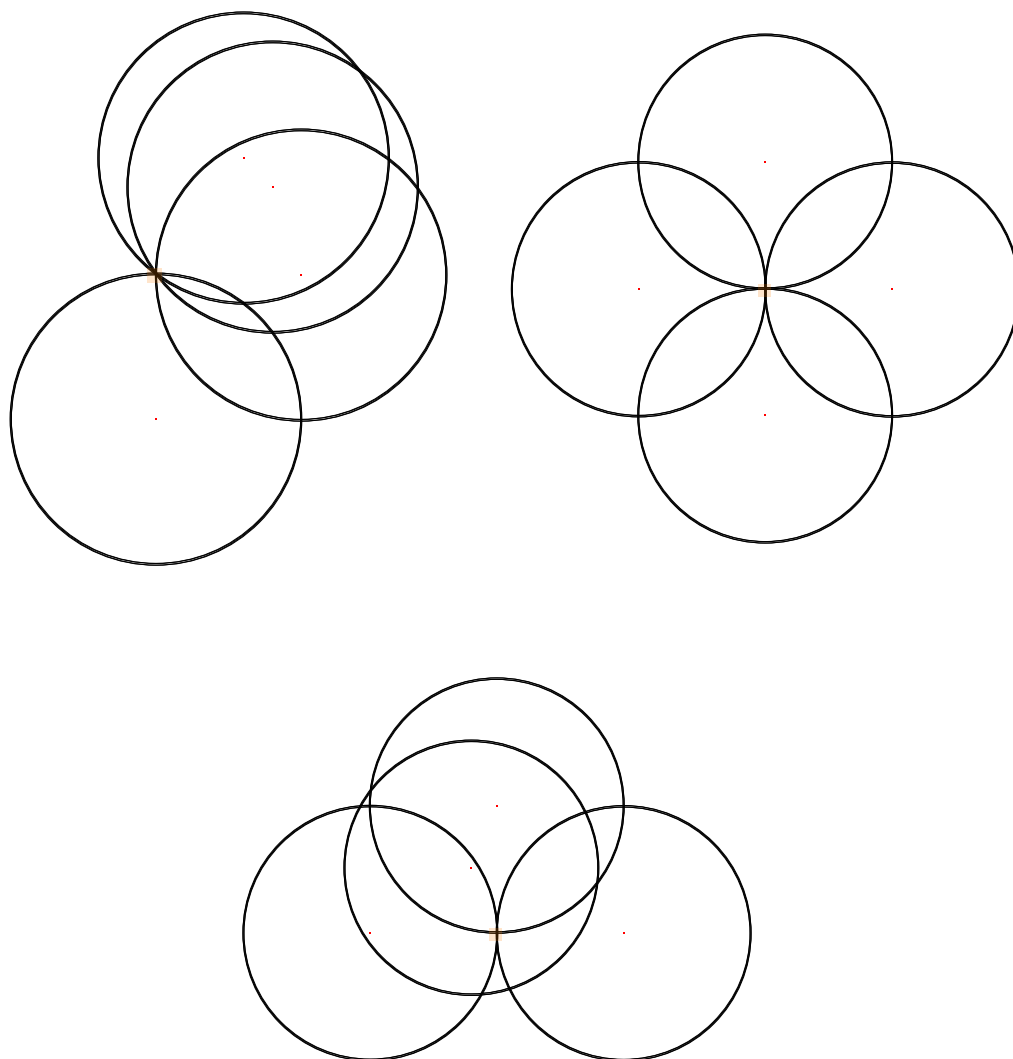


FIG. 2.4 – Vision globale des anneaux produits pour les trois jeux de mesures  $(S_1, S_2, S_3)$



simple, nous allons tout d'abord utiliser une approche assez naïve consistant à intersecter les solutions des systèmes bien contraints, que nous savons résoudre (voir section 1.3.3). Cette approche est équivalente au barycentre (voir section 2.5.1) mais en utilisant des intervalles.

**Intersection des solutions des systèmes bien contraints :** On calcule une boîte extérieure pour chaque système bien contraint et linéarisé avec le schéma de Newton, sur le même principe que le barycentre (en piochant deux équations parmi quatre), avec la méthode Hansen-Bliek.

Pour le jeu de mesures  $S_1$  on trouve les 6 boîtes :

- $(X_0, Y_0)_{12} = ([-1.12, -0.92], [-1.05, -0.85])$  ;
- $(X_0, Y_0)_{13} = ([-1.12, -0.92], [-1.05, -0.85])$  ;
- $(X_0, Y_0)_{14} = ([-1.12, -0.92], [-1.0491602, -0.9123423])$  ;
- $(X_0, Y_0)_{23} = ([-1.12, -0.92], [-1.0092368, -0.9352257])$  ;
- $(X_0, Y_0)_{24} = ([-1.12, -0.92], [-1.0469295, -0.9144853])$  ;
- $(X_0, Y_0)_{34} = ([-1.0886389, -0.92], [-1.0450276, -0.9158876])$ .

L'intersection de ces 6 boîtes donne un encadrement :

$$(X_0, Y_0)_{out} = ([-1.0886389, -0.92], [-1.0092368, -0.9352257])$$

Pour le jeu de mesures  $S_2$  :

$$(X_0, Y_0)_{out} = ([-1.0333455, -0.9663686], [-1.0427758, -0.9272658])$$

Pour le jeu de mesures  $S_3$  :

$$(X_0, Y_0)_{out} = \emptyset$$

**Avec la pseudo-inverse :** La méthode de la pseudo-inverse présentée à la section 2.3 est appliquée à notre problème complet (2.5.2) afin de définir une boîte extérieure.

Pour le jeu de mesures  $S_1$  :

$$(X_0, Y_0)_{out} = ([-1.12, -0.92], [-1.05, -0.85])$$

Pour le jeu de mesures  $S_2$  :

$$(X_0, Y_0)_{out} = ([-1.0469576, -0.95278], [-1.05, -0.9163203])$$

Pour le jeu de mesures  $S_3$  :

$$(X_0, Y_0)_{out} = \emptyset$$

**Avec les équations de Bjorck :** La méthode utilisant la notion de reste proposée par Bjorck et présentée à la section 2.3 est appliquée à notre problème complet (2.5.2) afin de définir une boîte extérieure.

Pour le jeu de mesures  $S_1$  :

$$(X_0, Y_0)_{out} = ([-1.0912703, -0.9244788], [-1.05, -0.85])$$

Pour le jeu de mesures  $S_2$  :

$$(X_0, Y_0)_{out} = ([-1.0437633, -0.9561531], [-1.05, -0.9238072])$$

Pour le jeu de mesures  $S_3$  :

$$(X_0, Y_0)_{out} = \emptyset$$

Avec le jeu  $S_3$ , les deux méthodes proposant un encadrement des solutions aux moindres carrés conduisent à un ensemble vide. Cela est dû à la mesure aberrante, qui “tire” les solutions aux moindres carrés hors de l’estimée initiale. On en conclut que si ces méthodes peuvent, en présence de mesure aberrante, produire une solution aux moindres carrés dans certains cas, elles ne sont pas robustes à ce phénomène.

### 2.6.2.2 Boîtes intérieures $\subset \Sigma_{\forall, \exists}$

**Avec la pseudo-inverse :** La méthode de la pseudo-inverse présentée à la section 2.3 est appliquée à notre problème complet (2.5.2) afin de définir une boîte intérieure.

Pour le jeu de mesures  $S_1$  :

$$(X_0, Y_0)_{in} = ([-1.0419312, -0.9677014], [-1.0340079, -0.9231524])$$

Pour le jeu de mesures  $S_2$  :

$$(X_0, Y_0)_{in} = ([-1.0322345, -0.9667117], [-1.0387352, -0.9329535])$$

Pour le jeu de mesures  $S_3$  :

$$(X_0, Y_0)_{in} = \emptyset$$

**Avec les équations de Bjorck :** La méthode utilisant la notion de reste proposée par Bjorck et présentée à la section 2.3 est appliquée à notre problème complet (2.5.2) afin de définir une boîte intérieure.

Pour le jeu de mesures  $S_1$  :

$$(X_0, Y_0)_{in} = ([-1.0078899, -1.0078799], [-0.9720849, -0.9720749])$$

Pour le jeu de mesures  $S_2$  :

$$(X_0, Y_0)_{in} = ([-1.0000132, -1.0000032], [-0.9894592, -0.9894492])$$

Pour le jeu de mesures  $S_3$  :

$$(X_0, Y_0)_{in} = \emptyset$$

## 2.7 Discussion et conclusion

Une nouvelle manière de répondre à un problème d'identification de paramètres a été introduite et appliquée à l'étalonnage d'un mécanisme simple. Les résultats sont présentés à la figure 2.5.

En utilisant des intervalles pour la représentation des incertitudes, l'approche du reste en utilisant les équations de Bjorck permet une approximation plus étroite de l'encadrement extérieur que l'approche de la pseudo-inverse. Elle fournit également un point (avec la boîte intérieure) qui est plus proche de la solution idéale que les approches classiques (type moindres carrés) dans un cas bien conditionné (le jeu  $S_2$ ) tout en certifiant les propriétés évoquées précédemment. Les méthodes décrites dans ce chapitre permettent donc une identification de paramètres efficace et avec des propriétés intéressantes, qui a été validée sur un exemple simple. Ces méthodes vont nous servir au chapitre 5 à étalonner le robot parallèle à câbles CoGiRo, ce qui constitue le principal objectif de ce manuscrit.

Néanmoins, nous avons vu que les méthodes par intervalles présentées ne sont pas robustes aux mesures aberrantes qui ont un effet non négligeable sur la solution. De plus, elles fournissent un résultat moins bon que l'approche de l'intersection deux à deux, même si cette dernière est combinatoire et entraîne un calcul coûteux.

*Remarque : Dû à la sur-évaluation causée par la pseudo-inverse, la boîte intérieure semble correspondre à l'encadrement le plus fin de l'enveloppe extérieure. Ceci constitue un point à éclaircir dans une étude ultérieure à ce manuscrit.*

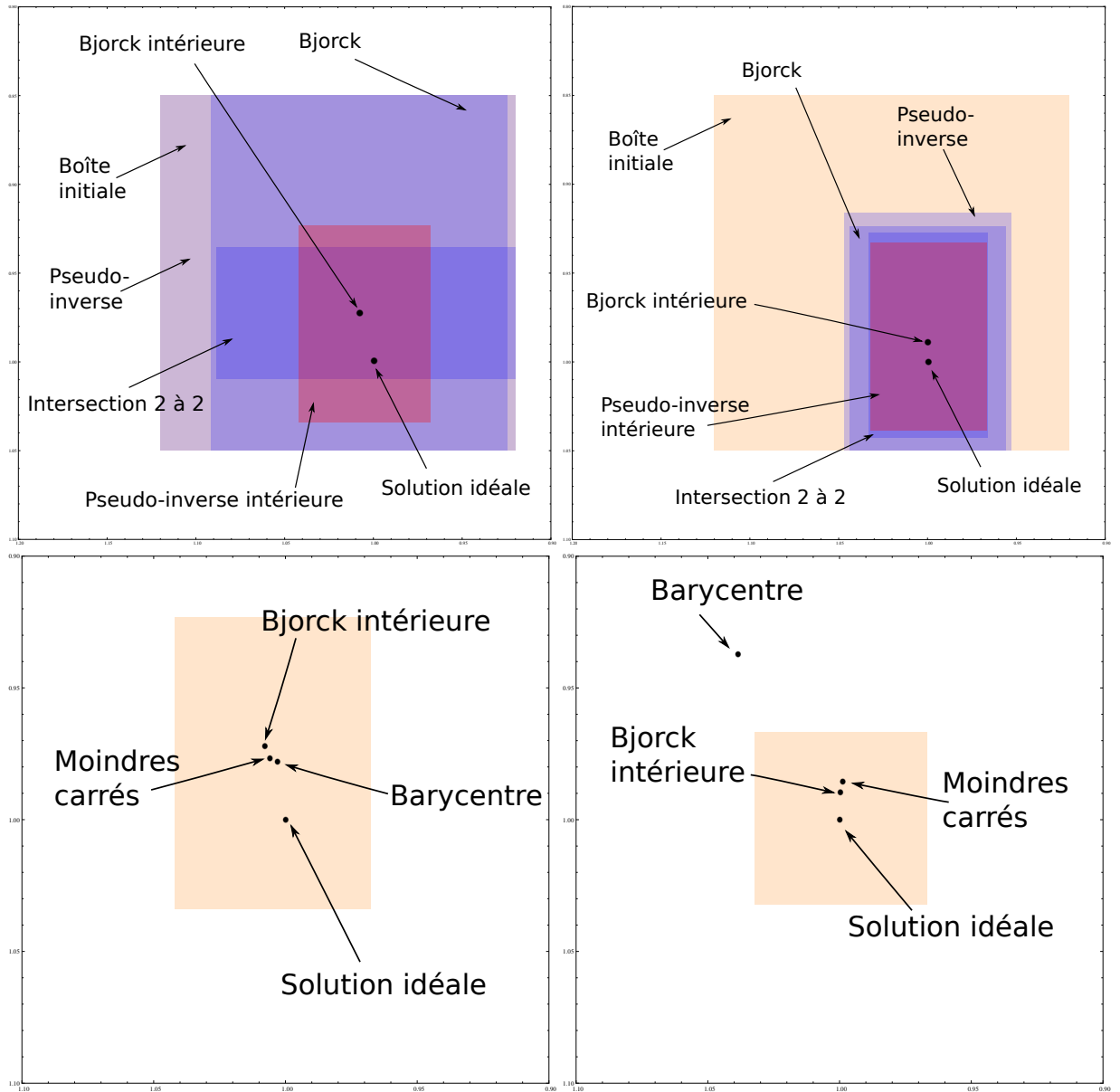


FIG. 2.5 – Résultats pour les boîtes extérieures en haut, pour les solutions ponctuelles en bas (représentées avec la meilleure boîte extérieure obtenue), avec le jeu  $S_1$  à gauche et le jeu  $S_2$  à droite.

**Visualisation du résultat pour le mécanisme :** Le dessin d'un cercle dont le centre est une boîte est un problème complexe, pour cela nous utilisons une méthode d'inversion ensembliste présentée dans [Jaulin 2010].

Les résultats sont donnés en terme d'anneau solution centré en  $(X_0, Y_0)$  (vision du mécanisme), et en terme de confirmation de satisfaction aux contraintes par un zoom sur l'intersection des anneaux centrés sur les mesures (vision étalonnage).

Les meilleurs résultats ont été obtenus avec Bjorck-Hansen-Bliek et avec Bjorck-Beaumont sur le jeu de mesures  $S_2$ , nous visualisons donc les résultats de ces deux méthodes aux figures 2.6 et 2.7.

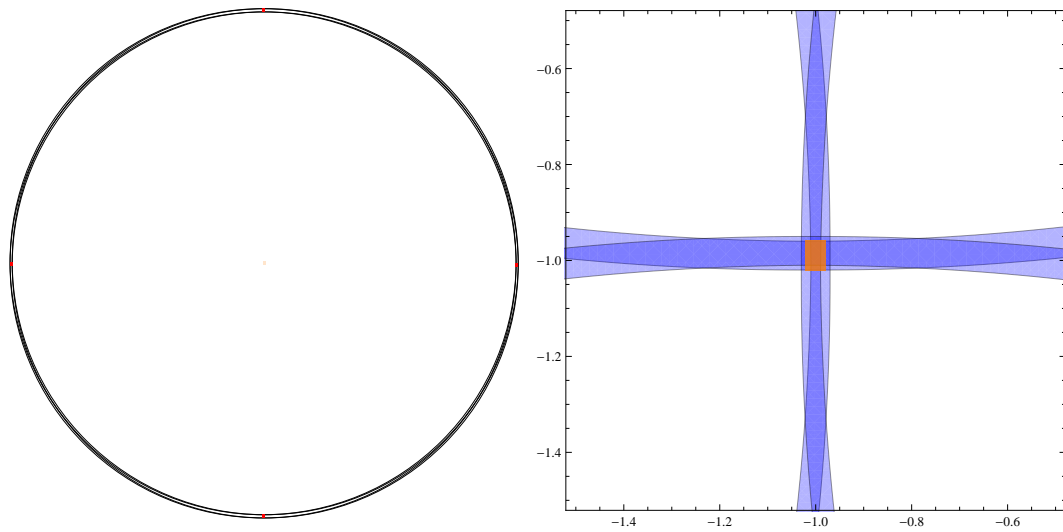


FIG. 2.6 – Résultat du calcul de  $[x]_{out}$  avec Newton-Bjorck-Hansen-Bliek pour le jeu  $S_2$

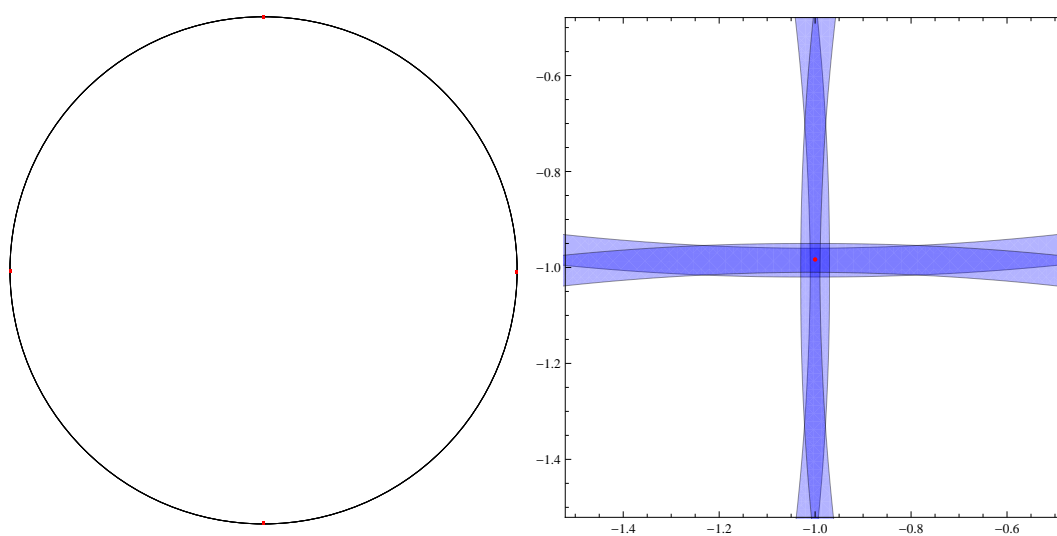


FIG. 2.7 – Résultat du calcul de  $[x]_{in}$  avec Newton-Bjorck-Beaumont pour le jeu  $S_2$



# Modélisation des robots à câbles

## Sommaire

<b>3.1</b>	<b>Modèle d'un câble</b>	<b>62</b>
3.1.1	Câble indéformable de masse nulle	63
3.1.2	Câble élastique de masse nulle	63
3.1.3	Câble élastique de masse non nulle	64
<b>3.2</b>	<b>Modèle géométrico-statique</b>	<b>65</b>
3.2.1	Modèle géométrique implicite	65
3.2.2	Transmission des efforts pour les modèles de câble non pesant	65
3.2.3	Transmission des efforts pour le modèle de câble pesant	67
3.2.4	Modèle géométrique complet et réaliste	67
3.2.5	Choix du modèle pour l'étalonnage	68
<b>3.3</b>	<b>Simplification du modèle de câble</b>	<b>69</b>
3.3.1	Vérification de l'hypothèse	69
3.3.2	Incertitude du modèle de câble	73
3.3.3	Espace de travail	74
<b>3.4</b>	<b>Conclusion</b>	<b>77</b>

Pour étalonner les robots à câbles, il est tout d'abord nécessaire d'obtenir les modèles qui définissent ce manipulateur (voir section 1.2.1.5). En ce qui concerne les manipulateurs que nous considérons, une des difficultés majeures repose sur la modélisation des câbles. De façon similaire aux robots parallèles, le modèle géométrique des robots à câbles consiste à relier la position et l'orientation de l'organe terminal aux distances entre les points d'attache des segments. Cependant, dans un modèle réaliste qui considère la masse et l'élasticité des câbles, la distance entre le point d'attache sur la plate-forme et le point de sortie du câble sur le bâti est fonction des caractéristiques du câble (masse et élasticité), de la longueur de câble déroulée et de la tension dans le câble. Ce modèle réaliste proposé par Irvine est décrit à la section 3.1.3. La longueur des câbles étant fonction des tensions et donc des efforts exercés sur la plate-forme, il est nécessaire de considérer le modèle statique du robot.

Un modèle géométrico-statique peut ainsi être défini pour les robots parallèles à câbles [Carricato 2012b]. Ce modèle complet est développé à la section 3.2.2. Néanmoins, nous verrons que la complexité de ce modèle et surtout le manque d'informations fiables sur les tensions dans les câbles ne permettent pas un étalonnage du robot en utilisant un tel modèle (voir section 3.2.5).

Il est donc nécessaire de simplifier ce modèle au moyen d'approximations qui permettent de découpler la statique de la géométrie. Sous certaines conditions que nous verrons à la section 3.3, la distance entre les extrémités des câbles est suffisamment proche de la longueur déroulée pour accepter cette simplification. La masse et l'élasticité du câble ont donc une influence négligeable sur cette distance qui alors



peut être considérée comme indépendante de la tension. Dans ce cas, un étalonnage géométrique est possible.

Dans le cas des grues à grande dimension manipulant de lourdes charges, la simplification du modèle du câble peut ne pas être valable. Dans ce chapitre, nous proposons une méthode pour vérifier si l'hypothèse de câble sans masse et sans élasticité est négligeable par rapport à un certain critère (section 3.3). Nous verrons également qu'il est possible de calculer et de garantir un sous espace de travail géométrique dans lequel cette hypothèse est acceptable. Dans ce sous espace de travail, on peut ainsi modéliser le robot d'une manière purement géométrique, tout en considérant l'incertitude sur la longueur de câble due à la simplification de son modèle.

### 3.1 Modèle d'un câble

Une source potentielle d'incertitude dans la modélisation des robots à câbles est le comportement des câbles sous tension. Actuellement, la majorité des travaux réalisés sur les robots parallèles à câbles utilisent un modèle de câble simplifié, sans masse ni élasticité, comme décrit dans la section 3.1.1. Cette modélisation permet en effet d'utiliser de nombreux résultats issus des robots parallèles rigides d'architecture de type Gough [Varziri 2007, Borgstrom 2009b].

Cependant dans certains cas (en particulier pour les robots de grande dimension), la masse propre et l'élasticité introduisent un fléchissement du câble dont le profil s'écarte alors de la ligne droite théorique. [Bouchard 2008] a présenté les conditions nécessaires pour pouvoir négliger le fléchissement et modéliser les câbles comme des segments droits. Pour cela, il compare le fléchissement au départ d'un câble réaliste avec la direction du câble si il était parfaitement droit. Cette étude est intéressante pour déterminer l'effet de la pesance du câble, mais ne considère pas l'effet de l'élasticité.

Si on relie un point  $A_i$  du bâti et un point  $B_i$  de la plate-forme avec un câble (voir figure 3.1), il est possible de définir plusieurs modèles de câble en fonction de la prise en compte ou non de la masse et de l'élasticité pour calculer la longueur de câble nécessaire afin de relier ces deux points.

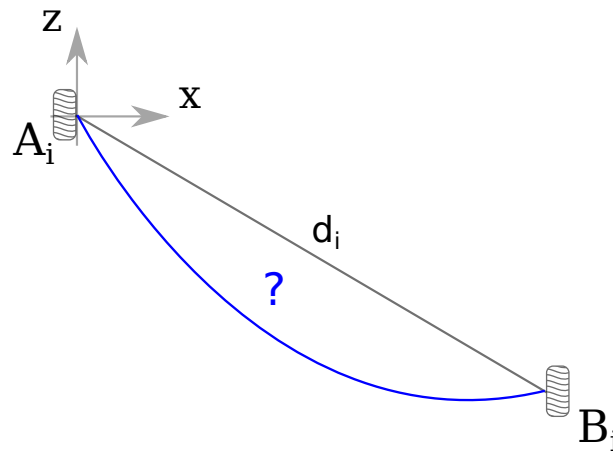


FIG. 3.1 – Comment modéliser un câble dans le plan vertical contenant  $A_i$  et  $B_i$  ?

Dans la suite nous noterons  $d_i$  la distance entre  $A_i$  et  $B_i$ ,  $l_i$  la longueur de câble déroulée (c'est à dire sans effort, à tension nulle), et  $L_i$  la longueur réelle du câble. Nous présentons différents modèles de câbles.

### 3.1.1 Câble indéformable de masse nulle

Le câble indéformable de masse nulle est un corps rigide qui n'existe pas. Toutefois, cette hypothèse forte permet une modélisation simplifiée des robots à câbles et l'erreur commise par l'utilisation de ce modèle est souvent considérée comme négligeable dans la littérature.

Ce modèle conduit à considérer le câble comme un segment droit de longueur  $l_i$ , c'est à dire la longueur déployée, reliant les points de sortie  $A_i$  et d'attaches  $B_i$ . On écrit donc  $d_i = l_i = L_i$  (voir figure 3.2).

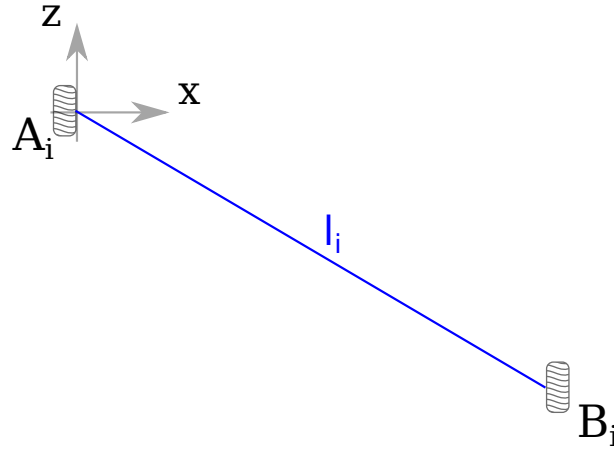


FIG. 3.2 – Un câble indéformable de masse nulle

### 3.1.2 Câble élastique de masse nulle

Tous les câbles sont plus ou moins élastiques, et peuvent donc s'étirer en fonction de la tension. Il est possible de prendre en compte cette élasticité tout en considérant le câble comme un segment droit, c'est à dire de masse considérée comme négligeable. Cependant, prendre en compte l'élasticité rend les longueurs des câbles fonction de la tension dans chaque câble [Merlet 2008b].

Ainsi la tension dans un câble de longueur sous effort  $L_i$  est donnée par la relation :

$$\tau_i = k(L_i - l_i) \quad (3.1)$$

où  $k$  définit les caractéristiques physiques du câble (la raideur linéique).

Ce modèle conduit à considérer le câble comme un segment droit de longueur  $L_i = d_i$ , fonction de la pose du robot et de la statique (figure 3.3).

Ce modèle est utile dans une configuration pleinement contrainte où les tensions peuvent être importantes et également lorsqu'une grande longueur de câble est soumise à effort [Merlet 2008b]. Cependant, pour de grandes longueurs de câbles dont la masse n'est pas négligeable, ou lorsque tous les câbles ne sont pas soumis à une tension importante, ce modèle n'est pas suffisamment réaliste.

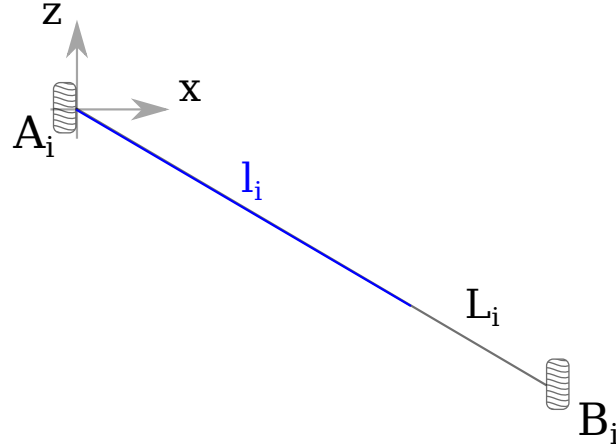


FIG. 3.3 – Un câble élastique de masse nulle

### 3.1.3 Câble élastique de masse non nulle

Un modèle de câble connu et un des plus réalistes est issu des travaux d'Irvine dans le génie civil [Irvine 1974]. Il est utilisé dans la modélisation de structure à câbles (pour les ponts à haubans principalement), et très rarement pour les robots parallèles à câbles (voir par exemple [Kozak 2006] sur la statique du robot Fast).

Dans le modèle de Irvine, le câble peut donc pendre et s'étirer, et les quantités  $l_i$ ,  $L_i$  et  $d_i$  sont trois grandeurs différentes (voir figure 3.4).

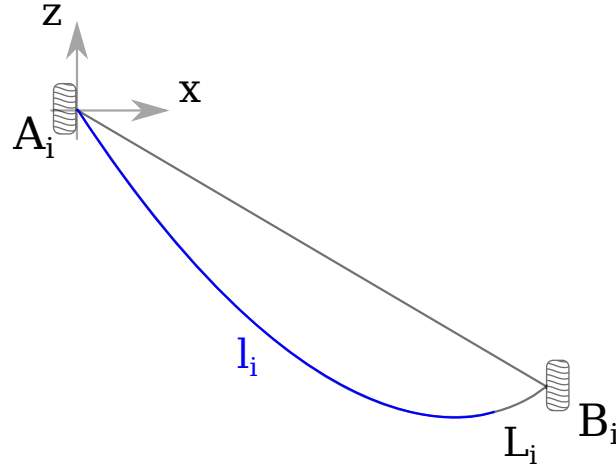


FIG. 3.4 – Un câble élastique de masse non nulle

On représente souvent le câble réaliste comme pendant pour symboliser le fait que sa masse est considérée, mais si le câble subit une tension suffisante, son profil est très proche d'une droite avec toutefois  $L_i \neq l_i$  (figure 3.3).

Pour un câble, les équations sont exprimées dans le plan défini par  $A_i$ ,  $B_i$  et la gravité, en considérant les paramètres géométriques et statiques de la configuration et les caractéristiques du câble :

- Les points d'attache  $A_i$  (sur le bâti) et  $B_i$  (sur la plate-forme)

- Câble : la masse linéique  $m$ , la raideur linéique (aussi appelée constante de rappel)  $k$  et la longueur sans effort (déroulée à vide)  $l_i$
- Les tensions appliquées :  $T_b = (T_b^x T_b^z)$  en  $B_i$  et  $T_a = (T_a^x T_a^z)$  en  $A_i$

Ainsi Irvine propose un système de 3 équations qui relie la longueur de câble déroulée  $l_i$ , la distribution des tensions en  $B_i$ , la position du point  $B_i$  et la norme de la tension  $\|T_a\|$  :

$$\begin{cases} B_i^x = \frac{T_b^x l_i}{k} + \frac{|T_b^x|}{mg} [\sinh^{-1}(\frac{T_b^z}{T_b^x}) - \sinh^{-1}(\frac{T_b^z - mgl_i}{T_b^x})] \\ B_i^z = \frac{mgl_i^2}{k} (\frac{T_b^z}{mgl_i} - \frac{1}{2}) + \frac{1}{mg} [\sqrt{T_b^{x2} + T_b^{z2}} - \sqrt{T_b^{x2} + (T_b^z - mgl_i)^2}] \\ \|T_a\| = \sqrt{T_b^{x2} + (T_b^z - mgl_i)^2} \end{cases} \quad (3.2)$$

Les équations de ce modèle le rendent évidemment difficile à manipuler.

## 3.2 Modèle géométrico-statique

Comme nous l'avons vu dans la section précédente, le modèle de câble le plus réaliste et complet implique que la longueur du câble est fonction de la tension à laquelle il est soumis. En conséquence, son utilisation dans les modèles géométriques impose de prendre en compte la statique du robot comme cela a été fait dans [Carricato 2012b].

### 3.2.1 Modèle géométrique implicite

Le modèle géométrique implicite d'un manipulateur parallèle à  $n = 6$  degrés de liberté (par exemple la plate-forme de Gough, voir figure 1.2) est décrit dans [Merlet 1997]. Il peut être utilisé pour modéliser la géométrie des manipulateurs parallèles à câbles [Barrette 2000]. Dans la figure 3.5 un manipulateur à 8 câbles et en configuration pleinement contrainte est utilisé pour définir le modèle géométrique implicite.

La plate-forme mobile, liée au repère  $\Omega_C$ , est connectée au bâti, liée au repère  $\Omega_O$ , par  $m = 8$  câbles. Le  $i^{eme}$  câble relie le point  $A_i$  sur le bâti (de coordonnées  $\mathbf{a}_i$  dans  $\Omega_O$ ) au point  $B_i$  sur la plate-forme mobile (de coordonnées  $\mathbf{b}_i$  dans  $\Omega_C$ ). La pose de la plate-forme mobile est définie par le vecteur de position  $\mathbf{p} = [x, y, z]$  et la matrice d'orientation  $\mathbf{R}_{3 \times 3}$  de  $\Omega_C$  exprimé dans  $\Omega_O$ .

Le modèle géométrique implicite peut alors s'écrire :

$$\|\mathbf{p} + \mathbf{R}\mathbf{b}_i - \mathbf{a}_i\| - d_i = 0, \quad i = [1...m] \quad (3.3)$$

où  $d_i$  est la distance entre les points  $A_i$  et  $B_i$  (qui sera la longueur minimale que pourra avoir le câble).

### 3.2.2 Transmission des efforts pour les modèles de câble non pesant

Un câble dont la masse est négligée (voir sections 3.1.2 et 3.1.1) reliant un point de sortie  $A_i$  et un point d'attache  $B_i$  exerce une force de traction de  $B_i$  vers  $A_i$  dépendant de la tension dans le câble  $\tau_i$ . La direction de cette force est

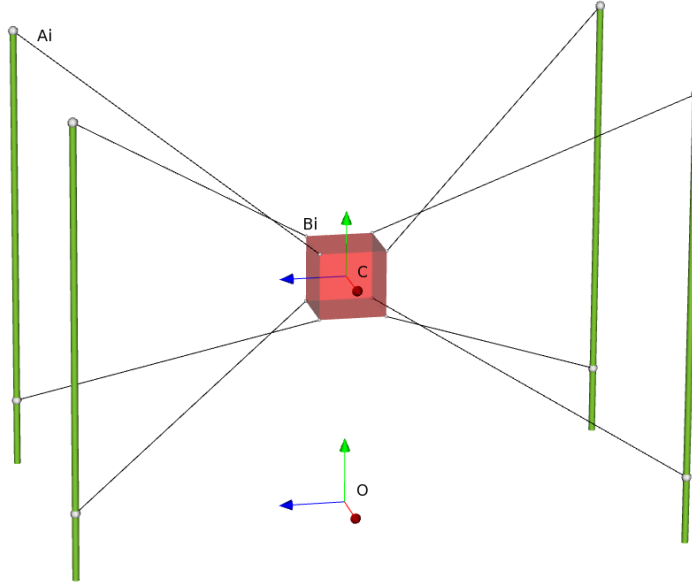


FIG. 3.5 – Un manipulateur parallèle à câbles

notée  $\mathbf{u}_i = \frac{\mathbf{p} + \mathbf{R}\mathbf{b}_i - \mathbf{a}_i}{d_i}$ . Le torseur d'effort (une force et un moment) ainsi engendré sur la plate-forme en son point de référence (origine de  $\Omega_C$ ) est donné par  $\tau_i \mathbf{w}_i$ , où  $\mathbf{w}_i$  s'écrit :

$$\mathbf{w}_i = \begin{bmatrix} \mathbf{u}_i \\ \mathbf{R}\mathbf{b}_i \times \mathbf{u}_i \end{bmatrix} \quad (3.4)$$

À noter que  $\mathbf{w}_i$  dépend de la pose de la plate-forme.

Les efforts extérieurs, comme le poids de la plate-forme et de son chargement par exemple, appliqués sur la plate-forme en son point de référence sont rassemblés dans le torseur d'efforts  $\mathbf{f}_e$ .

L'équilibre statique peut donc être écrit comme dans [Roberts 1998, Bruckmann 2008a] :

$$\mathbf{f}_e + \mathbf{W}\boldsymbol{\tau} = 0 \quad (3.5)$$

où :

$$\boldsymbol{\tau} = [\tau_1, \tau_2, \dots, \tau_m]^T \quad (3.6)$$

et :

$$\mathbf{W} = [\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_m] \quad (3.7)$$

En utilisant l'équation (3.5), pour une pose donnée et en connaissant le torseur des efforts extérieurs, il est donc possible d'estimer les tensions  $\tau$ . Cependant, les câbles ne pouvant exercer qu'une tension positive, la solution  $\tau$  obtenue ne sera pas valide si elle contient un  $\tau_i$  négatif. Dans le cas des robots en configuration suspendue, il est démontré dans [Merlet 2012a] qu'il est, par exemple, impossible pour un robot à  $m$  câbles rigides attachés en un seul point d'avoir  $m > 3$  câbles

tendus. Cet auteur conjecture que ce résultat reste vrai dans un cas plus général comme le notre (le phénomène peut d'ailleurs être observé sur le robot du projet CoGiRo). Dans le cas des robots suspendus à  $m > n$  câbles, l'équilibre statique doit donc être vérifié pour chaque combinaison de  $n$  câbles parmi  $m$ .

### 3.2.3 Transmission des efforts pour le modèle de câble pesant

Dans le cas où la masse du câble est considérée (voir section 3.1.3), la direction de la force exercée sur la plate-forme par le câble  $i$  est fonction du profil du câble défini par les composantes de la tension en  $B_i$ .

Tout d'abord, il est nécessaire de se placer dans le plan dans lequel le modèle de câble est défini. Nous supposons en général que le repère du robot a son axe  $z$  défini suivant avec la verticale (colinéaire avec la gravité). Ainsi nous passons du repère du robot au plan du câble  $i$  par une rotation d'un angle  $\gamma_i = \arctan(\frac{u_i^y}{u_i^x})$  autour de l'axe  $z$  (avec  $u_i = \overrightarrow{A_i B_i}$ ).

$$\text{La matrice de rotation } Q_i = \begin{pmatrix} \cos \gamma_i & -\sin \gamma_i & 0 \\ \sin \gamma_i & \cos \gamma_i & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

Le torseur d'effort  $\tau_i \mathbf{w}_i$  définissant la transmission des efforts à la plate-forme par le câble  $i$  s'écrit alors :

$$\mathbf{w}_i [\tau_i^x \tau_i^z]^T = \begin{pmatrix} Q_i^1 & Q_i^3 \\ \mathbf{Rb}_i \times Q_i^1 & \mathbf{Rb}_i \times Q_i^3 \end{pmatrix} \begin{pmatrix} T_{b_i}^x \\ T_{b_i}^z \end{pmatrix} \quad (3.8)$$

où  $Q_i^1$  et  $Q_i^3$  sont respectivement la première et la troisième colonne de  $Q_i$ .

### 3.2.4 Modèle géométrique complet et réaliste

Le modèle complet du robot comporte donc trois sous modèles : la géométrie, la statique et le modèle de câble réaliste.

Le modèle du câble lie les coordonnées du point  $B_i$  dans le plan du câble à la statique et à la longueur déroulée :

$$B_i^x - \frac{T_b^x l_i}{k} + \frac{|T_b^x|}{mg} [\sinh^{-1}(\frac{T_b^z}{T_b^x}) - \sinh^{-1}(\frac{T_b^z - mgl_i}{T_b^x})] = 0 \quad (3.9)$$

$$B_i^z - \frac{mgl_i^2}{k} (\frac{T_b^z}{mgl_i} - \frac{1}{2}) + \frac{1}{mg} [\sqrt{T_b^{x2} + T_b^{z2}} - \sqrt{T_b^{x2} + (T_b^z - mgl_i)^2}] = 0 \quad (3.10)$$

Les équations géométriques du robot permettent d'exprimer le changement de repère du plan du câble au robot, en liant la pose du robot (position  $\mathbf{p}$  et orientation  $\mathbf{R}$ ) au modèle de câble :

$$\begin{pmatrix} B_i^x \\ 0 \\ B_i^z \end{pmatrix} - Q_i^{-1}(\mathbf{p} + \mathbf{Rb}_i - \mathbf{a}_i) = 0 \quad (3.11)$$

Et enfin, la transmission des efforts de (3.2.3) nous permet d'écrire l'équilibre statique de la plateforme :

$$\mathbf{f}_e + \mathbf{W}\tau = 0 \quad (3.12)$$

Le modèle géométrico-statique réaliste comporte ainsi en théorie  $2m + n$  équations.

#### 3.2.4.1 Modèle géométrico-statique inverse

Le modèle géométrico-statique inverse, noté *MGSI*, consiste à calculer les longueurs de câble à dérouler  $l_i$  pour atteindre une position et une orientation données. Dans le cas général, où l'on cherche à modéliser des câbles réalistes, il est en plus nécessaire de calculer les tensions dans les câbles au moyen du modèle de la statique. Les inconnues du problème géométrique inverse d'un robot à  $m$  câbles et  $n$  ddl sont :

$$(T_{b_1}^x \ T_{b_1}^z \ l_1 \ \dots \ T_{b_m}^x \ T_{b_m}^z \ l_m) \quad (3.13)$$

soit  $3m$  inconnues.

On dispose donc de  $2m + n$  équations pour calculer  $3m$  variables. Le problème inverse est donc sous-contraint dans notre cas ( $m > n$ ). Cependant, certains câbles peuvent ne pas être tendus en configuration suspendue et donc ne supporteront pas la plate-forme [Merlet 2012b]. La statique doit donc être résolue uniquement avec les câbles exerçant un effort sur la plate-forme. Si on suppose qu'il y en a  $n$  sur les  $m$ , alors la statique nous donne  $n$  équations pour  $n$  inconnues et le modèle de câble nous permet de calculer les  $n$  longueurs de câbles à dérouler, les autres longueurs devant respecter une inégalité du type  $l_i > d_i$  (avec  $d_i$  provenant de 3.3). La résolution du *MGSI* avec un modèle de câble réaliste n'est pour ces raisons pas encore complètement maîtrisée.

#### 3.2.4.2 Modèle géométrico-statique direct

Le modèle géométrico-statique direct, noté *MGSD*, consiste à calculer la position et l'orientation de la plate-forme en fonction de la longueur des câbles déroulées  $l_i$ . Dans le cas général, où l'on cherche à modéliser des câbles réalistes, il est nécessaire de calculer les tensions dans les câbles au moyen du modèle de la statique. Les inconnues du problème géométrique direct d'un robot à  $m$  câbles et  $n$  ddl sont :

$$(\mathbf{p} \ \mathbf{R} \ T_{b_1}^x \ T_{b_1}^z \ \dots \ T_{b_m}^x \ T_{b_m}^z) \quad (3.14)$$

soit  $n+2m$  inconnues. En théorie, le problème direct est donc bien contraint :  $n+2m$  inconnues pour  $2m+n$  contraintes. On ne dispose pas d'informations sur le nombre de câbles en tension pour les différentes solutions du *MGSD*. Il convient donc de résoudre le *MGSD* pour chaque configuration possible de câbles. La résolution du *MGSD* est encore un problème ouvert [Merlet 2012b].

### 3.2.5 Choix du modèle pour l'étalonnage

On a vu qu'il était nécessaire de calculer les tensions dans les câbles. Pour l'étalonnage, la mesure des tensions est technologiquement délicate et très bruitée. La mesure n'est donc pas une solution immédiate pour l'étalonnage d'un robot à câbles en utilisant une modélisation réaliste de la statique et des câbles [Miermeister 2012].

Dans ces conditions, il est raisonnable de se demander si l'utilisation d'un modèle de câble simplifié, n'étant pas fonction de la statique, peut être une solution acceptable. Pour cela, nous proposons :

- de vérifier la validité de ce modèle simplifié ;
- de considérer l’erreur que cette modélisation peut entraîner sur la géométrie du robot ;
- de calculer un espace de travail dans lequel ce modèle simplifié est acceptable.

### 3.3 Simplification du modèle de câble

L’hypothèse du câble sans masse ni élasticité est couramment faite sur le comportement physique des câbles des robots. Cette hypothèse forte peut entraîner une erreur non négligeable sur la modélisation du robot [Riehl 2009, Agahi 2010]. Cette hypothèse n’a pourtant que rarement été vérifiée, même si on peut citer [Bouchard 2008].

Cette section présente une méthode pour vérifier cette hypothèse pour un robot et une utilisation donnés. L’approche intervalle et les outils utilisés permettent en plus de définir un nouveau modèle de câble et un nouvel espace de travail du robot.

#### 3.3.1 Vérification de l’hypothèse

La problématique est de vérifier l’hypothèse réalisée sur les câbles pour un robot et une tâche donnés. Un robot est défini entre autre par son espace de travail  $\mathcal{W}_X$ , et une tâche par les forces qu’elle nécessite et donc par les tensions dans les câbles. Ainsi vérifier cette hypothèse revient à la vérifier dans tout l’espace de travail du robot et pour toute une plage de tension dans les câbles. Si l’hypothèse est vérifiée de cette manière, alors la simplification du modèle du câble et donc du robot sera valide quelle que soit sa pose.

##### 3.3.1.1 Méthode

Nous posons  $\sigma_i = |l_i - d_i|$ , l’erreur réalisée entre la longueur déroulée  $l_i$  nécessaire d’après le modèle réaliste 3.1.3 pour atteindre le point  $B_i$  et  $d_i$  la distance entre  $A_i$  et  $B_i$ , c’est à dire le modèle simplifié obtenu sous l’hypothèse 3.1.1.

La méthode consiste à vérifier que cette différence de modélisation  $\sigma_i$  reste en dessous d’un seuil d’acceptabilité  $\varepsilon$ . Ce seuil peut être défini par rapport à diverses grandeurs : la précision des enrouleurs, la précision finale attendue, mais aussi le bruit de mesure des capteurs.

En ce qui nous concerne, la précision d’enroulement est la grandeur choisie comme seuil. En effet, si par exemple un système d’enroulement a une tolérance de  $\pm 5$  centimètres sur la longueur de câble effectivement déroulée, chercher un modèle de câble précis au millimètre près n’a pas de sens. Le seuil  $\varepsilon$  est donc choisi en fonction de la précision d’enroulement, et l’hypothèse du câble sans masse et sans élasticité est considérée comme acceptable pour le câble  $i$  si  $\sigma_i \leq \varepsilon$ .

La position des  $m$  points  $B_i$  de la plate-forme sont fonction de la pose du robot  $X = (p, R)$ . Les coordonnées de  $B_i$  dans le repère du robot  $\Omega_O$  sont  $e_i = p + Rb_i$ .

**Trois cas** peuvent être distingués pour l’étude d’une pose  $X \in \mathcal{W}_X$  :

- Pour les  $m$  points  $e_i$  :  $\sigma_i \leq \varepsilon$ , alors l’hypothèse est vérifiée.
- Il existe un ou plusieurs points  $e_i$  tel que  $\sigma_i > \varepsilon$ , alors il se présente deux possibilités :



- L'étude de la statique nous montre que le câble  $i$  ne peut pas être tendu dans la pose  $X$  (quelque soit la configuration des câbles). C'est à dire que la tension supposée positive est en réalité négative. L'hypothèse ne peut être rejetée.
- L'étude de la statique nous montre qu'il existe une configuration dans laquelle le câble  $i$  est tendu, alors la probabilité pour que l'erreur de modélisation soit importante est non nulle et l'hypothèse n'est pas vérifiée.

L'étude de la statique peut être réalisée d'après la section 3.2.2. La vérification de ces trois cas peut donner lieu à un algorithme qui n'a pas été implémenté.

Notre approche est plus simple mais plus pessimiste et ne fait pas intervenir l'étude de la statique. Nous considérons ainsi que l'hypothèse est vérifiée pour **une** pose  $X \in \mathcal{W}_X$ , si pour les  $m$  points  $e_i : \sigma_i \leq \varepsilon$ . Nous définissons le sous ensemble des poses acceptables comme :

$$S_X = \{X \in \mathcal{W}_X : \sigma_i \leq \varepsilon, \forall i = 1..m\} \quad (3.15)$$

La vérification proposée consiste à valider l'hypothèse pour **toutes** les poses de l'espace de travail du robot.

Une condition suffisante est basée sur la vérification de l'espace dual de  $S_X$ , noté  $S_{!X}$  et défini comme suit :

$$S_{!X} = \{X \in \mathcal{W}_X : \sigma_i > \varepsilon, \exists i = 1..m\} \quad (3.16)$$

Ainsi, la vérification de  $S_{!X} = \emptyset$  implique par construction que l'hypothèse est acceptable dans tout l'espace de travail du robot  $\mathcal{W}_X$ .

On peut remarquer que  $S_X \cup S_{!X} \equiv \mathcal{W}_X$ .

### 3.3.1.2 Simplification du problème

Les  $m$  points  $B_i$  dépendants de la pose  $X = [p, R]$  du robot sont tous contenus dans un même parallélépipède, quel que soit  $X \in \mathcal{W}_X$  (voir figure 3.6). On définit ce parallélépipède comme un ensemble  $\mathcal{B} \supset \{e_i : e_i = p + Rb_i, \forall X \in \mathcal{W}_X, i = [1..m]\}$ . Par symétrie, le test d'un câble reliant un point  $A_i$  à chaque élément de l'ensemble  $\mathcal{B}$  permet la vérification des  $m$  câbles pour toutes les poses prises dans l'espace de travail  $\mathcal{W}_X$ .

Le modèle du câble présenté à la section 3.1.3 nécessite de se placer dans le plan vertical contenant le profil du câble au moyen d'une rotation autour de l'axe vertical. Le plus grand plan vertical de  $\mathcal{B}$  étant le plan diagonal vertical, noté  $\mathcal{W}_B$ , la vérification de l'hypothèse pour les points contenus dans  $\mathcal{W}_B$  suffit à vérifier tout le parallélépipède (par isotropie du modèle de câble ou symétrie par rotation).

En fin de compte, l'étude de l'espace de travail  $\mathcal{W}_X$  peut donc se réduire à celle du plan diagonal  $\mathcal{W}_B$  (voir figure 3.6).

Avec cette simplification, on définit de la même manière les sous ensembles où l'hypothèse est acceptable :

$$S_B = \{B \in \mathcal{W}_B : \sigma \leq \varepsilon\} \quad (3.17)$$

et le dual :

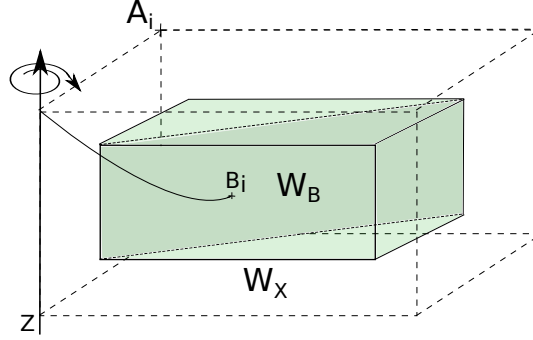


FIG. 3.6 – L'encombrement du robot en pointillés, son espace de travail  $\mathcal{W}_X$  et le plan diagonal  $\mathcal{W}_B$ .

$$S_{!B} = \{B \in \mathcal{W}_B : \sigma > \varepsilon\} \quad (3.18)$$

Par construction de la simplification, si un élément est trouvé dans  $S_{!B}$ , un élément existe dans  $S_{!X}$  et l'hypothèse n'est pas valide pour tout l'espace de travail.

De même on peut noter que  $S_B \cup S_{!B} \equiv \mathcal{W}_B$ . Il est plus facile de montrer que  $S_{!B} = \emptyset$  que de vérifier que  $S_B \equiv \mathcal{W}_B$ . Ainsi prouver que  $S_{!B} = \emptyset$  implique que  $\sigma \leq \varepsilon$  est vérifié pour toutes les poses de l'espace de travail du robot.

### 3.3.1.3 Approche adaptée aux intervalles

On l'a vu dans la section 1.3, les intervalles sont adaptés aux problèmes nécessitant de considérer tout un espace de recherche, et permettent de calculer des ensembles, ou tout du moins des sur-estimations de ces ensembles.

L'analyse par intervalles et les méthodes présentées au chapitre 1 peuvent donc nous permettre de répondre à notre problématique qui est de vérifier si  $S_{!B} = \emptyset$ .

Pour cela nous calculons une boîte sur-estimant  $S_{!B}$ , que l'on note  $\square S_{!B}$ , avec un système de contraintes défini par les équations (3.2) et la contrainte supplémentaire  $\sigma > \varepsilon$ . Si le calcul nous donne  $\square S_{!B} = \emptyset$ , alors on peut conclure que  $S_{!B} = \emptyset$  (l'arithmétique par intervalles nous permet de certifier que  $S_{!B} \subset \square S_{!B}$ ) et donc que l'hypothèse est acceptable pour tout l'espace de travail du robot.

Nous n'utilisons pas directement les équations (3.2), mais un système avec 5 variables auxiliaires  $(c, d, e, h, j)$  et des contraintes supplémentaires permettant une résolution plus rapide. Il a en effet été montré dans [Araya 2008] comment l'exploitation des sous-expressions communes (par exemple, ici, la sous-expression remplacée par  $j$  qui apparaît deux fois dans le système d'équations) permet dans certaines conditions d'améliorer la contraction par propagation de contraintes. Le système de contraintes définissant  $S_{!B}$  est finalement :

$$\left\{ \begin{array}{l} mgB_x - eT_b^x + T_b^x \ln((h-c)(c-d+j)) = 0 \\ mgB_z - T_b^x h - T_b^z e + \frac{ke^2}{2} + T_a = 0 \\ T_a - T_b^x j = 0 \\ cT_b^x - T_b^z = 0 \\ h - \sqrt{(c^2) + 1} = 0 \\ j - \sqrt{((c-d)^2) + 1} = 0 \\ dT_b^x - mgl = 0 \\ ek - mgl = 0 \\ |l - \sqrt{B_x^2 + B_z^2}| > \varepsilon \end{array} \right. \quad (3.19)$$

La résolution de ce système de contraintes est réalisée avec l'algorithme de *Branch & Contract* et les outils présentés dans la section 1.3.

Une phase de tests a montré que la meilleure stratégie est d'utiliser le contracteur 3BCID avec 2000 tranches, la monotonie des contraintes avec l'algorithme de propagation Mohc et enfin l'heuristique de bisection **Smea**r.

### 3.3.1.4 Résultats expérimentaux

La méthode a été appliquée aux deux robots étudiés dans cette thèse (présentés dans le Chapitre 1) et les résultats sont présentés ci dessous.

**ReelAx8 :** Pour ce robot (voir annexe B), le plan diagonal est défini de telle sorte que  $B^x = [1, 3.7]$  et  $B^z = [1, 2]$ , la plage de tension par câble est considérée de 20N à 120N et le seuil est fixé à  $\varepsilon = 5mm$ . L'outil rend une boîte vide pour  $S_B$ . Ceci implique que l'hypothèse est acceptable pour ce robot sur tout son espace de travail. Ainsi l'utilisation du modèle de câble simplifié est valide et permet d'utiliser directement différents résultats de la robotique parallèle classique.

**CoGiRo :** Pour ce robot (voir annexe C), le plan diagonal est défini de telle sorte que  $B^x = [1, 17]$  et  $B^z = [1, 5]$ , la plage de tension par câble est considérée de 40N à 5000N et le seuil est fixé à  $\varepsilon = 5cm$ . L'outil renvoie une solution immédiatement. L'hypothèse n'est donc pas vérifiée et son utilisation pourrait entraîner des erreurs importantes sur la modélisation. Ceci signifie qu'un modèle plus complexe doit être implémenté ou que l'erreur induite par la simplification se doit d'être prise en compte.

### 3.3.1.5 D'autres informations obtenues par optimisation

En complément de ce résultat sur la validité de l'hypothèse, les outils à notre disposition peuvent nous permettre d'enrichir la connaissance du robot. Plus particulièrement, l'optimisation globale avec l'algorithme de *Branch & Bound* présenté en section 1.3.6.4 rend possible le calcul de certaines bornes. Par exemple, il est possible de calculer la tension  $T_b$  minimale afin que l'hypothèse soit acceptable (câble suffisamment tendu). On peut également calculer l'erreur  $\sigma$  maximale (en valeur

absolue) commise sur l'espace de travail. Cette erreur est positive lorsque le câble est pendant et négative lorsque le câble est étiré.

**ReelAx8 :** L'analyse sur le robot ReelAx8, pour lequel l'hypothèse a été vérifiée, donne des résultats confirmant cette validation.

- $T_a$  minimale pour conserver  $|\sigma| < \varepsilon$ , voir Table 3.1, colonne 1 ;
- $\sigma$  maximale pour  $T_a = 20N$ , voir Table 3.1, colonne 2 ;
- $\sigma$  minimale pour  $T_a = 120N$ , voir Table 3.1, colonne 3 ;
- $T_a$  maximale pour conserver  $|\sigma| < \varepsilon$ , voir Table 3.1, colonne 4.

	1	2	3	4
Erreur $\sigma$ (m)	0.005	<b>0.0006</b>	<b>-0.0037</b>	-0.005
Tension $T_a$ (N)	<b>1.4</b>	20	120	<b>171.5</b>
Temps calc. (s)	200	3	5	30

TAB. 3.1 – Résultats (en gras) obtenus pour le robot ReelAx8 avec une méthode d'optimisation globale

Ces résultats nous montrent que pour l'espace de travail donné, il est possible de faire varier les tensions, que l'on peut en partie contrôler car ce robot est en configuration pleinement contrainte, de  $1.4N$  à  $171.5N$  tout en conservant un modèle de câble que l'on a jugé acceptable.

**CoGiRo :** La même analyse a été réalisée sur le robot CoGiRo et elle confirme le rejet de l'hypothèse.

- $\sigma$  maximale pour  $T_a = 40N$ , voir Table 3.2, colonne 1 ;
- $T_a$  minimale pour conserver  $|\sigma| < \varepsilon$ , voir Table 3.2, colonne 2 ;
- $T_a$  maximale pour conserver  $|\sigma| < \varepsilon$ , voir Table 3.2, colonne 3 ;
- $\sigma$  minimale pour  $T_a = 1000N$ , voir Table 3.2, colonne 4 ;
- $\sigma$  minimale pour  $T_a = 5000N$ , voir Table 3.2, colonne 5.

	1	2	3	4	5
Erreur $\sigma$ (m)	0.05	<b>0.047</b>	-0.05	<b>-0.057</b>	<b>-0.281</b>
Tension $T_a$ (N)	<b>38.9</b>	40	<b>878</b>	1000	5000
Temps calc. (s)	300	3	5133	5	5

TAB. 3.2 – Résultats (en gras) obtenus pour le robot CoGiRo avec une méthode d'optimisation globale

Ces informations nous permettent de définir une plage d'utilisation du robot permettant la simplification du modèle de câble tout en limitant l'erreur commise. On voit également que l'allongement des câbles est très important lorsqu'ils sont soumis à une tension élevée. L'élasticité ne peut donc pas être négligée si le robot manipule des charges de 500kg. Au contraire, l'effet de la pesanteur est assez limité si on parvient à maintenir une tension dans le câble, même relativement faible.

### 3.3.2 Incertitude du modèle de câble

L'optimisation globale peut nous permettre de calculer l'erreur maximale  $\sigma_{max}$  (positive si le câble est pendant) et l'erreur minimale  $\sigma_{min}$  (négative si le câble est

étiré). Ainsi on peut écrire ce résultat sous forme d'intervalle  $[\sigma] = [\sigma_{min}, \sigma_{max}]$ . Ce qui pour chaque pose de l'espace de travail et pour chaque câble conduit à la relation d'inclusion :

$$l_i - d_i \in [\sigma], \quad \forall i = [1..m], \forall X \in \mathcal{W}_X \quad (3.20)$$

Cette relation implique que l'on est capable d'encadrer le modèle le plus réaliste 3.1.3 par le modèle simplifié 3.1.1 auquel on a ajouté l'erreur maximale que ce modèle entraîne.

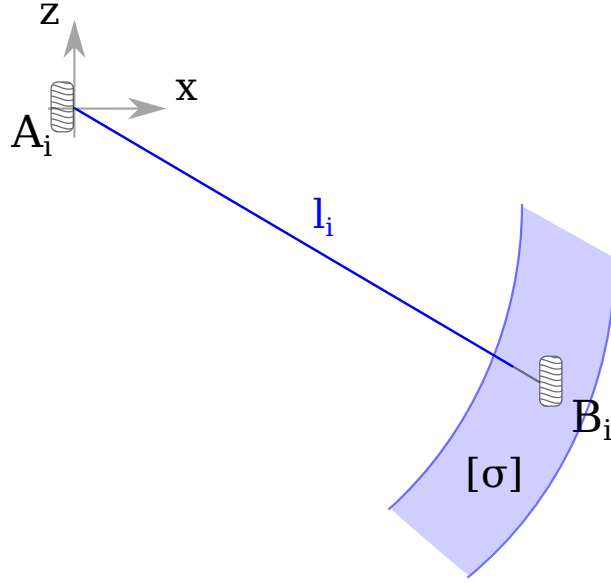


FIG. 3.7 – Un câble simplifié et l'erreur

Ainsi, même si ce modèle de câble est simple, il est garanti de contenir le câble réaliste. On réécrit le modèle géométrique implicite (3.3) sous forme d'intervalles avec ce nouveau modèle de câble ( $d_i = l_i - \sigma$ ) :

$$\|\mathbf{p} + \mathbf{R}\mathbf{b}_i - \mathbf{a}_i\| - (l_i - [\sigma]) = 0, \quad i = [1..m] \quad (3.21)$$

Ce modèle est sur-estimé mais il borne l'incertitude due à la modélisation des câbles pour tout l'espace de travail du robot et toute la plage de tension possible. Nous verrons au chapitre 5 que ce modèle peut être utilisé pour étalonner un robot à câbles.

### 3.3.3 Espace de travail

Si l'hypothèse a été validée pour le robot ReelAx8 pour tout son espace de travail, elle a été rejetée pour le robot CoGiRo. Toutefois, les outils d'analyse par intervalles mis en place permettent de définir un sous espace de travail dans lequel l'hypothèse est acceptable (s'il existe).

Cet espace de travail a été défini précédemment (voir 3.3.1.1) comme étant :

$$S_X = \{X \in \mathcal{W}_X : \sigma_i \leq \varepsilon, \forall i = 1..m\} \quad (3.22)$$

On note  $\mathcal{SW}_X$  un sous espace de travail où l'hypothèse est valide (il n'est pas forcément optimal). C'est à dire que l'on peut écrire :

$$X \in \mathcal{SW}_X \Rightarrow \sigma_i \leq \varepsilon, \forall i \in [1..m] \quad (3.23)$$

et également que  $\mathcal{SW}_X \subset S_X \subseteq \mathcal{W}_X$ .

En utilisant la même approche que pour la vérification de l'hypothèse, nous allons passer par le dual et vérifier pour une boîte  $[x] \subset \mathcal{W}_X$  :

$$S_{!}[x] = \{x \in [x] : \sigma > \varepsilon\} = \emptyset \quad (3.24)$$

De même il est préférable de ramener le problème à l'étude d'une diagonale couvrant  $[x]$ , que l'on note  $[b]$  (voir section 3.3.1.2).

### 3.3.3.1 Recherche d'un espace de travail

Un algorithme simple est utilisé (3.3.3.1). Il fournit une boîte dans laquelle l'hypothèse est acceptable. Il consiste à calculer itérativement  $S_{!}[b]$  sur la diagonale ( $[b]$ ) d'un espace de travail de taille décroissante ( $[x]$ ). L'espace de travail est pour cela réduit sur chaque dimension de manière symétrique (car le robot est quasi symétrique) mais arbitraire. Il s'agit en quelque sorte d'un rognage des bornes de l'espace de travail, et ce jusqu'à ce que  $S_{!}[b] = \emptyset$ .

---

#### Algorithme 3 Algorithme pour définir un $\mathcal{SW}_X$

---

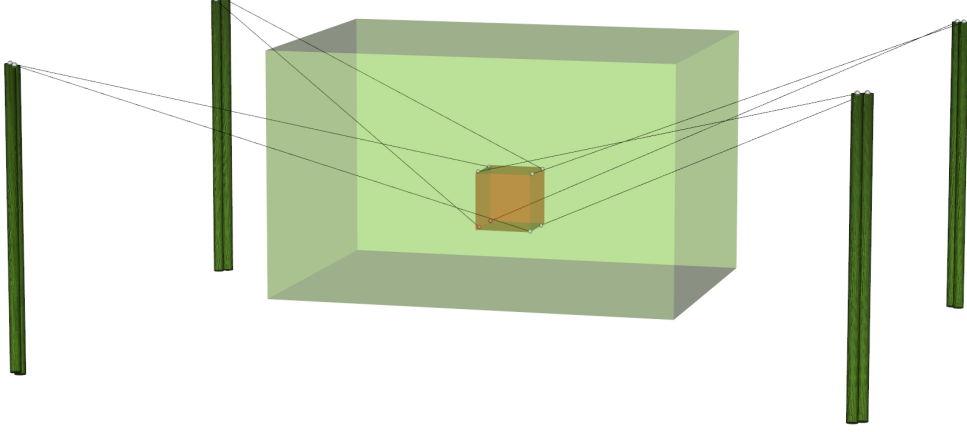
```

 $[x] = \mathcal{W}_X$ 
Calcul de  $S_{!}[b] = \{b \in [b] : \sigma > \varepsilon\}$ 
tant que  $S_{!}[b] \neq \emptyset$  faire
    Réduire  $[x]$ 
    Calcul de  $S_{!}[b] = \{b \in [b] : \sigma > \varepsilon\}$ 
 $\mathcal{SW}_X = [x]$ 

```

---

Le résultat de cet algorithme appliqué au robot CoGiRo est présenté à la figure 3.8. La plate-forme peut ainsi se déplacer dans la boîte  $[-3.65, 3.66] \times [-2.63, 2.68] \times [1, 2.74]$  tout en autorisant l'utilisation d'un modèle de câble simplifié.

FIG. 3.8 – Le robot et le sous espace  $\mathcal{SW}_X$ 

### 3.3.3.2 Pavage de l'espace de travail

Un autre algorithme peut fournir un espace de travail plus précis. Il consiste à approximer l'ensemble  $S_X$  par une liste de boîtes plus ou moins grandes contenues dans  $S_X$ , c'est à dire dans lesquelles l'hypothèse est acceptable. Cette liste de boîtes est définie par  $\mathcal{L}_{S_X}$  à l'algorithme 3.3.3.2.

---

#### Algorithme 4 Algorithme pour paver $S_X$

---

On ajoute  $\mathcal{W}_X$  à la liste  $\mathcal{L}$

On initialise le résultat du pavage  $\mathcal{L}_{S_X} = \emptyset$

**répéter**

On pioche  $[x]$  un élément de  $\mathcal{L}$

**si**  $w([x]) > \text{seuil}$  **alors**

Calcul de  $S_{[b]} = \{b \in [b] : \sigma > \varepsilon\}$

**si**  $S_{[b]} = \emptyset$  **alors**

On ajoute  $[x]$  à la liste  $\mathcal{L}_{S_X}$

**sinon**

On bissecte  $[x]$  sur une dimension (avec une heuristique) en  $[x]_1$  et  $[x]_2$

On ajoute  $[x]_1$  et  $[x]_2$  à la liste  $\mathcal{L}$

**sinon**

La boîte  $[x]$  est trop petite alors on l'oublie.

**jusqu'à**  $\mathcal{L} \neq \emptyset$

---

Ainsi, chaque élément de  $\mathcal{L}_{S_X}$  est un sous espace de travail où l'hypothèse est acceptable. Cet algorithme n'a pas été mis en œuvre et peut donner lieu à un développement futur.

## 3.4 Conclusion

L'objectif de ce manuscrit étant d'étalonner un robot à câbles, les modèles pouvant définir ce type de manipulateur ont été étudiés. Le modèle réaliste du comportement d'un câble, en considérant sa masse et son élasticité, nécessite de connaître la tension dans ce câble. Cette information n'est accessible ni par le calcul en présence d'incertitudes, ni par la mesure qui se trouve être inexploitable dans notre cas. L'utilisation d'un modèle de câble simplifié, négligeant la masse et l'élasticité, est donc nécessaire pour l'étalonnage.

Dans ce chapitre, nous avons vu une approche servant à vérifier l'erreur induit par cette simplification, entre la longueur du câble déployée et la distance entre ses points d'attache. En effet, si cette erreur due à la modélisation est négligeable au regard de la précision d'enroulement, alors on se ramène à un cas classique de robot parallèle. Dans le cas contraire, nous verrons au chapitre 5 qu'il est possible de considérer cette erreur due à la modélisation comme une incertitude et que l'étalonnage reste possible en utilisant des méthodes de l'analyse par intervalles.





# Étalonnage des robots parallèles à câbles

## Sommaire

<b>4.1</b>	<b>Modélisation</b>	<b>80</b>
4.1.1	Étalonnage	81
4.1.2	Auto-étalonnage	82
<b>4.2</b>	<b>Identifiabilité</b>	<b>84</b>
4.2.1	Identifiabilité dans la littérature	84
4.2.2	Quelques définitions	85
4.2.3	Recherche de poses optimales	86
4.2.4	Identifiabilité après la prise de mesure	87
<b>4.3</b>	<b>Expérimentations</b>	<b>89</b>
4.3.1	Mesures	89
4.3.2	Procédure d'étalonnage du prototype ReelAx8	91
4.3.3	Procédure d'étalonnage du robot CoGiRo	95
<b>4.4</b>	<b>Discussion</b>	<b>100</b>

La commande des robots parallèles à câbles nécessite de bien connaître les paramètres du modèle choisi pour approximer au plus près le comportement du robot. Pour cela, nous avons à notre disposition une estimation initiale des paramètres, issue généralement de la conception. La précision de positionnement de l'organe terminal (ou de la plate-forme) est directement fonction de la finesse du modèle choisi, de la connaissance des paramètres de ce modèle ainsi que de la qualité des capteurs.

- Le modèle utilisé dans une commande est un compromis entre l'approximation d'une réalité physique et le type d'informations disponibles (les mesures) pour le renseigner (par exemple nous ne considérons pas la statique, voir section 3.2.5) ;
- L'incertitude sur les paramètres initiaux du robot est fonction de sa qualité de fabrication et d'assemblage ;
- La qualité des informations provenant des capteurs proprioceptifs dépend du budget qui leur est alloué, de la technologie disponible ainsi que de la connaissance de leur paramètres intrinsèques.

L'utilisation d'un modèle, issu d'un compromis et approximant la réalité plus ou moins fidèlement, implique une différence entre ce modèle et d'autres modèles pouvant être plus réalistes. Cette différence est appelée erreur ou incertitude de modélisation.

Les incertitudes de modélisation, sur les paramètres ou les capteurs, doivent être diminuées, prises en compte ou limitées dans leurs effets si nous voulons améliorer la

précision d'un robot. Quelques principes ont été vus à la section 1.2 afin d'améliorer la précision d'un robot manipulateur.

Dans notre cas, la partie conception et construction du robot ne sont pas dans notre champs d'action. Le robot CoGiRo (voir figure 1.11) a été conçu de manière à améliorer sa précision globale d'après [Riehl 2011], il utilise un réseau de caméras pour mesurer la pose et les départs des câbles [Dallej 2012] et sa construction est de qualité industrielle.

Notre tâche est de proposer des méthodes d'étalonnage géométrique du robot CoGiRo, afin d'atteindre une précision importante au regard de la dimension du robot (inférieure à 5cm) et de qualifier le robot pour son utilisation. L'étalonnage géométrique d'un robot consiste à réduire l'incertitude - améliorer la connaissance - par l'identification des paramètres géométriques de celui ci.

Afin de réaliser un étalonnage performant, spécialement dans un cas difficile comme celui des robots parallèles à câbles, il est nécessaire de procéder à neuf étapes (voir section 1.2.1) :

1. Choisir un modèle pour le robot (en général celui utilisé par la commande) ;
2. Ajouter éventuellement des capteurs ou des contraintes mécaniques afin d'obtenir des informations redondantes sur l'état interne du robot ;
3. Modéliser les équations d'étalonnage ;
4. Étudier l'identifiabilité théorique des paramètres de ces équations ;
5. Chercher des configurations où réaliser les mesures pour un étalonnage performant ;
6. Acquérir des mesures ;
7. Vérifier l'identifiabilité expérimentale des paramètres ;
8. Identifier des paramètres ;
9. Valider les résultats de l'identification.

Une distinction est possible sur la provenance des mesures nécessaires à l'étalonnage. Si ces mesures peuvent être fournies par les capteurs utilisés par la commande (codeurs des moteurs, système de caméras pour la commande, etc), on parle alors d'auto-étalonnage. L'auto-étalonnage n'est possible que dans le cas d'un robot redondant (en actionnement ou en mesure). S'il est nécessaire de recourir à un appareil de mesure externe (un laser par exemple), on parle alors simplement d'étalonnage.

Dans ce chapitre, ces deux types d'étalonnage sont appliqués aux robots étudiés et nous allons voir qu'il est possible de spécialiser certaines des étapes de l'étalonnage aux robots parallèles à câbles. La partie identification est faite avec la contrainte d'employer des méthodes classiques, fréquemment utilisées et disponibles.

## 4.1 Modélisation

Le modèle géométrique d'un robot à câbles est semblable à celui des robots parallèles classiques (voir le chapitre 3).

Le modèle géométrique implicite peut donc s'écrire (avec les définitions de la section 3.2.1) :

$$\|\mathbf{p} + \mathbf{R}\mathbf{b}_i - \mathbf{a}_i\| - d_i = 0, \quad i = [1...m] \quad (4.1)$$

En considérant les câbles comme des segments sans masse et non déformables, alors la distance entre les points de sortie  $A_i$  sur le bâti et les points d'attache  $B_i$  sur la plate-forme est considérée égale à la longueur de câble déployée :  $d_i = l_i$ .

Ainsi le modèle géométrique inverse (*MGI*) devient :

$$l_i = \|\mathbf{p} + \mathbf{R}\mathbf{b}_i - \mathbf{a}_i\|, \quad i = [1...m] \quad (4.2)$$

Les équations d'étalonnage (ou de fermeture) sont les  $m$  systèmes de  $N_C$  équations suivantes (avec les définitions de la section 1.2.1.5) :

$$\begin{aligned} f_{k,i}(M_k, V) &= \|\mathbf{p}_k + \mathbf{R}_k \mathbf{b}_i - \mathbf{a}_i\| - l_{i,k}, \quad k = [1...N_C], i = [1...m] \\ &= 0 \end{aligned}$$

#### 4.1.1 Étalonnage

Nous disposons d'un appareil de mesure externe (un laser dans notre cas) afin de faire des mesures de la pose (position et orientation). Le système d'étalonnage à résoudre est donc défini par les fonctions  $f_{k,i}(M_k, V)$  avec les données suivantes :

- Les mesures  $\mathbf{M}_k = [l_{i,k}, \mathbf{p}_k, \mathbf{R}_k]$
- Les paramètres géométriques  $\xi = [\mathbf{a}_i, \mathbf{b}_i]$
- Les variables à identifier  $\mathbf{V} = [\Upsilon, \xi] = [\emptyset, \xi] = \xi$

Ici,  $\Upsilon = \emptyset$  car toutes les variables considérées sont soit mesurées soit des paramètres d'étalonnage.

Les  $m$  systèmes sont indépendants et il est donc possible d'étalonner le robot câble par câble. Pour le  $i^{eme}$  câble, le système d'équations est défini par :

$$\begin{cases} f_{1,i}(M_1, \xi) = \|\mathbf{p}_1 + \mathbf{R}_1 \mathbf{b}_i - \mathbf{a}_i\| - l_{i,1} \\ \vdots \\ f_{N_C,i}(M_{N_C}, \xi) = \|\mathbf{p}_{N_C} + \mathbf{R}_{N_C} \mathbf{b}_i - \mathbf{a}_i\| - l_{i,N_C} \end{cases} \quad (4.3)$$

Ce système peut s'écrire sous une forme vectorielle  $F = \{f_{1,i}, \dots, f_{N_C,i}\}$ . Une solution aux moindres carrés pour identifier les paramètres  $\mathbf{a}_i$  et  $\mathbf{b}_i$  de ce système minimisera par exemple le critère  $F^T F$  (voir section 1.2 pour d'autres critères).

---

#### Algorithme 5 Identification des paramètres du câble $i$

---

**tant que**  $\Delta F > \varepsilon$  **faire**  
 $\xi_i \leftarrow IS(\xi_i, M)$

---

Le schéma d'identification *IS* consiste à résoudre itérativement le système linéarisé  $J_{\xi_i} \cdot \Delta \xi_i = \Delta F$ . Avec

$$J_{\xi_i} = 2 \begin{pmatrix} \frac{-(\mathbf{p}_1 + \mathbf{R}_1 \mathbf{b}_i - \mathbf{a}_i)^T}{d_{1,i}} & \frac{(\mathbf{R}_1^T (\mathbf{p}_1 + \mathbf{R}_1 \mathbf{b}_i - \mathbf{a}_i))^T}{d_{1,i}} \\ \vdots & \vdots \\ \frac{-(\mathbf{p}_{N_C} + \mathbf{R}_{N_C} \mathbf{b}_i - \mathbf{a}_i)^T}{d_{N_C,i}} & \frac{(\mathbf{R}_{N_C}^T (\mathbf{p}_{N_C} + \mathbf{R}_{N_C} \mathbf{b}_i - \mathbf{a}_i))^T}{d_{N_C,i}} \end{pmatrix}_{(N_C \times n)} \quad (4.4)$$

Ce système linéaire est résolu avec par exemple une *SVD* (voir section 1.2.2.2).

### 4.1.2 Auto-étalonnage

L'hypothèse faite sur les câbles permet de considérer le robot comme redondant (voir section 1.2.1), c'est à dire que nous disposons de plus d'informations ( $m$  câbles donc  $m$  contraintes) qu'il est nécessaire pour déterminer la pose de la plate-forme (définie par  $n$  degrés de liberté) :  $m > n$ . Il est alors possible d'étalonner le robot sans avoir recours à un appareil de mesure externe.

Le système d'étalonnage à résoudre est donc défini par les fonctions  $f_{k,i}(M_k, V)$  avec les données suivantes :

- Les mesures  $\mathbf{M}_k = [l_{i,k}]$
- Les paramètres géométriques  $\xi = [\mathbf{a}_i, \mathbf{b}_i]$
- Les variables à identifier  $\mathbf{V} = [\Upsilon, \xi] = [\mathbf{p}_k, \mathbf{R}_k, \xi]$

Avec ces données, la jacobienne  $J_V$  des équations (1.3) est composée de la jacobienne d'identification (la même que celle de l'étalonnage) et de la jacobienne cinématique inverse  $J_\Upsilon$ .

Une des difficultés de l'auto-étalonnage est d'éliminer les variables  $\tilde{\Upsilon}_k = [\mathbf{p}_k, \mathbf{R}_k]$  (qui n'est pas le résultat recherché) du vecteur à identifier  $V = [\tilde{\Upsilon}_k, \xi]$  [Daney 2000].

Dans [Patel 2000], l'auteur estime les variables de pose  $\tilde{\Upsilon}_k = [\mathbf{p}_k, \mathbf{R}_k]$  à chaque itération de l'algorithme d'identification avec un modèle géométrique direct itératif (*MGD*). Ce modèle géométrique est fonction de l'estimée des paramètres géométriques (en cours d'identification) obtenue à l'itération précédente. Une fois la pose estimée avec  $n$  contraintes, il utilise les  $m - n$  contraintes restantes pour identifier les paramètres géométriques.

---

**Algorithme 6** Identification avec un modèle géométrique direct

---

```

tant que  $\Delta f > \varepsilon$  faire
   $[p, R]_{k+1} \leftarrow MGD(\tilde{\xi}_k, [\tilde{p}, \tilde{R}]_k, M)$ 
   $\xi_{k+1} \leftarrow IS(\tilde{\xi}_k, [\tilde{p}, \tilde{R}]_{k+1}, M)$ 

```

---

où  $IS$  est un schéma d'identification, typiquement un schéma de type itératif basé sur les moindres carrés, permettant de résoudre le système linéarisé avec une approche de type Taylor (1.3).

L'étape du *MGD* utilise une approximation des paramètres  $\xi$  pour déterminer la pose du robot  $[p, R]$ , et ainsi ni l'existence d'une solution, ni la convergence ne sont alors garanties.

Avec cette approche instable numériquement, une itération de l'algorithme d'étalonnage peut ne pas améliorer la connaissance des paramètres géométriques, mais au contraire la dégrader. Ce phénomène a été observé dans notre cas [Sandretto 2012a]. Pour cette raison nous proposons une **identification complète** qui consiste à calculer  $\Upsilon$  et  $\xi$  simultanément dans le processus d'identification.

---

**Algorithme 7** Identification complète

---

```

tant que  $\Delta f > \varepsilon$  faire
   $V_{k+1} \leftarrow IS(\tilde{V}_k, M)$ 

```

---

L'**identification complète** utilise seulement un schéma d'identification, la pose et les paramètres géométriques sont calculés en même temps et ainsi le problème de la convergence du *MGD* est éliminé.

Avec cette méthode les  $m$  systèmes ne sont plus indépendants et le système total, rangé par pose, est donné par :

$$\left\{ \begin{array}{ll} f_{1,1}(M_1, V) = & \|\mathbf{p}_1 + \mathbf{R}_1 \mathbf{b}_1 - \mathbf{a}_1\| - l_{1,1} \\ & \vdots \\ f_{1,m}(M_1, V) = & \|\mathbf{p}_1 + \mathbf{R}_1 \mathbf{b}_m - \mathbf{a}_m\| - l_{m,1} \\ f_{2,1}(M_2, V) = & \|\mathbf{p}_2 + \mathbf{R}_2 \mathbf{b}_1 - \mathbf{a}_1\| - l_{1,2} \\ & \vdots \\ f_{N_C,m}(M_{N_C}, V) = & \|\mathbf{p}_{N_C} + \mathbf{R}_{N_C} \mathbf{b}_m - \mathbf{a}_m\| - l_{m,N_C} \end{array} \right. \quad (4.5)$$

Les variables à identifier sont alors rangées dans le vecteur  $V$  comme suit :  $V = [a_1, b_1, \dots, a_m, b_m, X_1, \dots, X_{N_C}]_{(n.(m+N_C))}$ , où  $X$  représente la pose (position  $p$  et orientation  $R$ ) du mobile avec  $X = [x, y, z, \alpha \mathbf{v}]$ . Le vecteur d'orientation  $\delta = \alpha \mathbf{v}$  se compose de  $\mathbf{v}$  l'axe de rotation et de  $\alpha$  l'angle de rotation en radians (représentation de la vis ou de Rodrigues).

Le schéma d'identification  $IS$  consiste à résoudre itérativement le système linéarisé  $J_V \cdot \Delta V = \Delta F$

$$\text{où la Jacobienne } J_V = \begin{pmatrix} J_{\xi_1} & 0 & J_{X_1} & 0 \\ & \ddots & & \ddots \\ 0 & J_{\xi_m} & 0 & J_{X_{N_C}} \end{pmatrix}_{(mN_C \times n(m+N_C))}$$

Avec  $J_{X_k}$  les dérivées des équations d'étalonnage par rapport aux paramètres de pose. Ces matrices peuvent être approximées par la jacobienne cinématique inverse à la pose  $k$  [Merlet 2006]. En effet, pour un déplacement petit, le vecteur  $\Delta X$  est approximativement égale au torseur de vitesse de la plate-forme  $\dot{X}$ . La jacobienne cinématique inverse peut être obtenue avec les vecteurs normalisés de plücker [Merlet 2006]. En posant  $n_i = (\mathbf{p} + \mathbf{R}\mathbf{b}_i - \mathbf{a}_i)/d_i$  :

$$J_{inv} = \begin{pmatrix} n_1 & n_1 \times (\mathbf{R}\mathbf{b}_1) \\ \vdots & \vdots \\ n_m & n_m \times (\mathbf{R}\mathbf{b}_m) \end{pmatrix}_{(m \times n)} \quad (4.6)$$

On peut considérer que  $\Delta X \approx \dot{X}$ . La vitesse angulaire est définie par :

$$\Omega = \dot{\alpha} \mathbf{v} + \sin \alpha \dot{\mathbf{v}} + (1 - \cos \alpha) \mathbf{v} \times \dot{\mathbf{v}} \quad (4.7)$$

Une approximation de Taylor au 1<sup>er</sup> ordre est ensuite généralement utilisée [Merlet 2006], pour les  $\alpha$  petits, telle que :

$$\Omega \approx \dot{\alpha} \mathbf{v} + \alpha \dot{\mathbf{v}} = \dot{\delta} \quad (4.8)$$

Et donc  $\Delta X \approx \dot{X}$  et  $J_X \approx J_{inv}$ .

**Proposition 1** *L'utilisation d'un développement au second ordre de  $\Omega$  permet une approximation plus précise de la jacobienne  $J_X$ .*

**Démonstration 1** *Développement au second ordre de  $\cos \alpha \approx 1 - \frac{\alpha^2}{2}$  et de  $\sin \alpha \approx \alpha$ .*

L'approximation de  $\Omega$  devient :

$$\begin{aligned}\Omega_2 &= \dot{\alpha}\mathbf{v} + \alpha\dot{\mathbf{v}} + (1 - (1 - \frac{\alpha^2}{2}))\mathbf{v} \times \dot{\mathbf{v}} \\ &= \dot{\delta} + \frac{\alpha^2}{2}\mathbf{v} \times \dot{\mathbf{v}} \\ &= \dot{\delta} + \frac{1}{2} \cdot \alpha\mathbf{v} \times \alpha\dot{\mathbf{v}}\end{aligned}$$

On a  $\dot{\delta} = \dot{\alpha}\mathbf{v} + \alpha\dot{\mathbf{v}}$  et donc  $\alpha\dot{\mathbf{v}} = \dot{\delta} - \dot{\alpha}\mathbf{v}$

$$\begin{aligned}\Omega_2 &= \dot{\delta} + \frac{1}{2} \cdot \alpha\mathbf{v} \times (\dot{\delta} - \dot{\alpha}\mathbf{v}) \\ &= \dot{\delta} + \frac{\alpha}{2} \cdot \mathbf{v} \times \dot{\delta} \quad (\text{car } \mathbf{v} \times \mathbf{v} = 0)\end{aligned}$$

Et ainsi nous proposons une approximation plus précise de la jacobienne :

$$J_X \approx J_{inv} \cdot (I + M) \quad (4.9)$$

où  $M$  est le pré-produit vectoriel de  $\frac{\alpha}{2}\mathbf{v}$ .

## 4.2 Identifiabilité

Le principe d'étalonnage consiste dans l'utilisation d'un modèle et de son observation dans différents états pour identifier les paramètres le constituant. Toutefois, la définition du modèle ou les configurations utilisées peuvent ne pas permettre l'identification de tous les paramètres. Les raisons peuvent être diverses mais nous en listons 3 catégories :

- certains paramètres sont couplés identiquement quelques soient les états du robot considérés [Khalil 1991, Renaud 2002] ;
- certains paramètres ont un effet négligeable sur la variation du modèle ;
- certains paramètres ne sont pas suffisamment excités par le choix des états pris par le robot.

La détection de ces paramètres non identifiables est appelée l'*identifiabilité des paramètres*.

### 4.2.1 Identifiabilité dans la littérature

L'identification des paramètres  $\xi = \{\xi_1, \dots, \xi_p\}$  d'un modèle numérique  $M(X, \xi)$  est une approche connue, toutefois l'identifiabilité consistant à savoir ce que l'on peut identifier, un passage pourtant obligé, reste marginalement évoquée.

On peut cependant noter la méthode la plus utilisée, lorsque l'identifiabilité est vérifiée, qui consiste dans l'analyse de la matrice jacobienne d'identification ( $J_\xi$ ) [Nahvi 1996, Sun 2008].

Pour les mécanismes sériels, l'estimation de l'identifiabilité des paramètres a été obtenue en calculant le rang de  $J_\xi$  [Khalil 1991, Meggiolaro 2000]. Pour les mécanismes parallèles, Khalil et Besnard [Khalil 1999] ont appliqué une méthode similaire en analysant la matrice  $J_\xi$  pour également évaluer son rang.

En effet, si le rang de la matrice jacobienne est inférieur au nombre de paramètres à étalonner, cela indique qu'un ou plusieurs paramètres ne pourront être

identifiés. Une décomposition  $QR$  [Besnard 2001], permet de détecter les ensembles de paramètres liés.

Cette méthode a une limite. En effet, la valeur du rang est obtenue par un calcul numérique et peut mener à discussion. Les méthodes classiques utilisées pour calculer le rang d'une matrice consistent dans le comptage des valeurs singulières non-nulles. Les valeurs singulières pouvant être proche de zéro, il faut parfois les négliger, et la question se pose sur le seuil à utiliser.

Ces valeurs quasi-nulles font apparaître une catégorie de paramètres rarement considérés : les paramètres faiblement identifiables. Si il est relativement simple de trouver les paramètres identifiables ou non-identifiables, les paramètres faiblement identifiables sont difficilement discernables. Ces paramètres apparaissent justement lorsque certains paramètres sont liés à d'autres, ou lorsque leur influence sur le modèle est faible (la raison peut être numérique) [Ollivier 2000].

### 4.2.2 Quelques définitions

Ils existent plusieurs classes de paramètres définies pour l'identifiabilité. Ces définitions peuvent être explicitées de manière ensembliste [Little 2010], probabiliste [Jacquez 1985], sous forme de contraintes sur  $J_\xi$  [Khalil 1991] ou de dénombrabilité de solution à l'identification [Lagrange 2005].

Une définition basée sur l'univocité de l'identification est intéressante :

- Les paramètres observables : les paramètres qui apparaissent dans le modèle. C'est à dire les paramètres tels que si  $\xi$  prend les valeurs  $\xi^1$  et  $\xi^2$  distinctes, alors  $\exists \hat{X}$  tels que  $M(\hat{X}, \xi^1) \neq M(\hat{X}, \xi^2)$ .
- Les paramètres non observables : les paramètres dont la variation n'a aucun effet sur le modèle. Ces paramètres ne seront pas considérés dans la suite car ils correspondent à une erreur de modélisation.
- Les paramètres identifiables : les paramètres tels que si  $\xi$  prend les valeurs  $\xi^1$  et  $\xi^2$  distinctes alors  $M(\hat{X}, \xi^1) \neq M(\hat{X}, \xi^2)$ , et ce  $\forall \hat{X}$ .
- Les paramètres non identifiables : les paramètres tels que si  $\xi$  prend les valeurs  $\xi^1$  et  $\xi^2$  distinctes alors  $\exists \hat{X}$  tels que  $M(\hat{X}, \xi^1) = M(\hat{X}, \xi^2)$ .

Une autre approche est donnée dans un extrait de la thèse de Lagrange, inspirée de [Walter 1997], voici l'extrait :

*"Considérons un processus décrit par un modèle  $M(\cdot)$  dont on souhaite estimer les paramètres idéaux  $\xi^*$ . On se place dans le cadre idéal où les données sont non bruitées et où les entrées peuvent être choisies librement.*

- *Un paramètre  $\xi_i$  est identifiable si pour presque tout  $\xi^* \in P$  :*  

$$M(\hat{\xi}) = M(\xi^*) \Rightarrow \hat{\xi}_i = \xi_i^*$$
- *Un paramètre  $\xi_i$  est non-identifiable si il existe un ensemble non dénombrable de  $\hat{\xi}_i$  :  $M(\hat{\xi}) = M(\xi^*)$*
- *Un paramètre  $\xi_i$  est localement identifiable si il existe un ensemble dénombrable de  $\hat{\xi}_i$  :  $M(\hat{\xi}) = M(\xi^*)$  "*

Aux groupes de paramètres classiques, on peut ajouter les notions de globalement ou localement identifiable qui décrivent des sous ensembles (ou voisinages) de valeurs



de  $\xi$  dans lesquels  $\xi_i$  est identifiable. Ces notions de voisinage ne sont pas forcément intéressantes dans le cas d'un étalonnage car on suppose considérer suffisamment de mesures pour obtenir une solution unique au problème d'identification.

Une notion beaucoup plus importante pour nous et moins utilisée est la notion de faiblement identifiable.

Par interpolation et avec la même approche de dénombrabilité, cela revient à la définition :

- Un paramètre  $\xi_i$  est faiblement identifiable s'il existe un ensemble non vide de  $\hat{\xi}_i : M(\hat{\xi}) \simeq M(\xi^*)$

On voit alors que la notion de faiblement identifiable, en lien avec le comparateur  $\simeq$ , est subjective et reliée à la précision du calcul numérique [Ollivier 2000].

Cette classe de paramètre est donc peu souvent définie, toutefois Little [Little 2010] en donne une définition locale basée sur les probabilités :

- Un paramètre  $\xi_i$  est localement faiblement identifiable si il existe un voisinage ou la probabilité associée à l'observation  $X$  est maximisée par au plus un  $\xi$ .

### 4.2.3 Recherche de poses optimales

Pour améliorer la qualité et la robustesse d'un étalonnage, il existe plusieurs méthodes : augmenter le nombre de mesures ou en augmenter leur apport en terme de qualité numérique. Si la première méthode peut au premier abord sembler la mieux et la plus simple, en réalité le bruit de mesure fait qu'ajouter des mesures n'améliore pas le résultat [Nahvi 1996]. De plus, elle nécessite une longue phase de mesure qui peut s'avérer fastidieuse. Dans [Driels 1990] et [Daney 2005], il est démontré qu'il est préférable de choisir un jeu de mesure judicieusement plutôt que de multiplier les mesures.

L'approche classique afin de choisir un jeu de mesure le mieux possible pour l'étalonnage est de définir un critère à optimiser, puis d'utiliser un algorithme d'optimisation qui va tenter de trouver le jeu de mesure qui minimisera au mieux ce critère.

*Note : cette étape de recherche de poses optimales se fait classiquement en amont de la prise de mesures, mais certains l'ont faite après afin de choisir les meilleures poses parmi celles effectuées [Daney 2002, Takeda 2004].*

#### 4.2.3.1 Critère

Le critère de choix des configurations de mesures pour l'étalonnage est généralement donné par des indices calculés à partir des valeurs singulières de la matrice d'identification  $J_\xi$  (dérivés des équations d'étalonnage en fonction des paramètres géométriques à identifier, voir section 1.2.2.2). La liste, leurs effets et leur corrélation sont donnés dans [Hollerbach 1996]. Un autre critère utilisant l'étude de l'orthogonalité de la matrice d'identification est introduit dans [Takeda 2004].

Il existe aujourd'hui de nombreux critères dont une partie provient de la théorie des plans d'expériences optimaux [Sun 2008].

On peut citer quelques critères venant de la statistique :

- Le critère de D-optimalité qui maximise le déterminant de  $(J_\xi^T J_\xi)$  ;
- Le critère de A-optimalité qui minimise la trace de  $(J_\xi^T J_\xi)$  ;
- Le critère de E-optimalité qui maximise la plus petite valeur singulière de  $(J_\xi^T J_\xi)$ .

Et on peut citer les critères venant de l'étalonnage s'appuyant sur les valeurs singulières de  $J_\xi$  et notées  $\mu_1 \geq \dots \geq \mu_r$  :

- Maximiser l'index d'observabilité  $O_1 = \frac{\sqrt[r]{\mu_1 \dots \mu_r}}{\sqrt{N_C}}$  [Borm 1991]. Cet index est similaire à la D-optimalité.
- Minimiser le conditionnement  $O_2 = \frac{\mu_1}{\mu_r}$ .
- Maximiser la plus petite valeur singulière  $O_3 = \mu_r$ . Cet index est similaire à la E-optimalité.
- Maximiser l'index d'amplification du bruit  $O_4 = \frac{\mu_r^2}{\mu_1}$  [Nahvi 1996].

Tous les critères testés donnent sensiblement le même résultat, nous utiliserons donc le critère  $O_2$  qui est un des plus utilisés [Sun 2008].

#### 4.2.3.2 Algorithme

Différents algorithmes peuvent répondre à ce problème de choix de configuration. On peut citer les algorithmes de type *recuit simulé* ou d'*algorithme génétique* [Zhuang 1994], ou issus de la théorie des plans d'expériences optimaux [Daney 2005].

Un algorithme ne faisant pas appel à l'optimisation locale a été utilisé. Pour cela, nous échantillonnons l'espace de travail sur les 6 ddl (le pas peut être choisi en fonction de la précision du robot, du temps de calcul, etc) : ceci nous donne la liste  $\mathcal{L} = \{X_1, \dots, X_N\}$ . La liste des poses sélectionnées est initialisée avec 10 éléments de  $\mathcal{L}$  (les premières, les dernières ou choisies au hasard par exemple), 10 étant un nombre de poses suffisant pour étalonner le robot :  $\mathcal{Q} = \{X_i, \dots, X_j\}$ . La jacobienne d'identification est calculée avec les poses de  $\mathcal{Q}$  :  $J_\xi(\mathcal{Q})$ . Nous calculons le critère  $c = O_2(J_\xi)$ . Et ensuite, tant que l'algorithme améliore le critère, il essaie d'ajouter, d'enlever ou d'échanger des poses, ce qui revient à ajouter, enlever ou échanger des lignes de la jacobienne.

---

#### Algorithme 8 Algorithme de choix de poses

---

Échantillonner :  $\mathcal{L} = \{X_1, \dots, X_N\}$

Initialiser :  $\mathcal{Q} = \mathcal{L}(i, \dots, j)$  ( $nb(i, \dots, j) = 10$ )

Calculer la jacobienne  $J_\xi(\mathcal{Q})$

Calculer le critère  $O_2(J_\xi)$

**répéter**

    Ajouter à  $\mathcal{Q}$  un élément de  $\mathcal{L}$  s'il améliore le critère

    Enlever à  $\mathcal{Q}$  un élément de  $\mathcal{L}$  s'il ne dégrade pas le critère

    Échanger un élément de  $\mathcal{Q}$  avec un élément de  $\mathcal{L}$  s'il améliore le critère

**jusqu'à** ce que le critère ne s'améliore plus

---

Cet algorithme peut être lourd en temps de calcul et ne comporte pas d'optimisation locale, mais il fournit de bons résultats, proches de la théorie (aux bords de l'espace de travail à différentes orientations).

#### 4.2.4 Identifiabilité après la prise de mesure

L'étude de l'identifiabilité, nous l'avons évoqué, peut être une question à laquelle il est difficile de répondre simplement. En effet, cet aspect de l'étalonnage pose de nombreuses questions, à la fois théoriques et pratiques.

L'approche consistant à étudier la matrice Jacobienne d'identification est la plus simple. Mais la "qualité" de la matrice Jacobienne des paramètres n'est pas triviale à estimer.

À chaque étape de l'étalonnage le problème de l'identifiabilité intervient :

- Lors de la modélisation, l'homogénéité des paramètres, des variables et des équations d'étalonnage a une influence sur la normalisation de la matrice.
- Le choix des configurations du robot peut permettre d'améliorer l'orthogonalité et le conditionnement de la matrice (section 4.2.3).
- Une fois les mesures effectuées, l'identifiabilité peut être vérifiée en instanciant la matrice jacobienne avec les valeurs obtenues.

En effet, dû à certaines contraintes (espace de travail limité, faisabilité d'une des poses sélectionnées) et à la précision du manipulateur, les configurations choisies afin d'optimiser l'identifiabilité des paramètres peuvent ne pas être réalisées exactement. De plus, le facteur d'échelle entre les paramètres peut induire une identifiabilité numériquement faible pour un paramètre, en théorie identifiable [Ollivier 2000].

L'identifiabilité peut donc être à nouveau étudiée après la prise de mesure. Pour cela, une méthode classique a fait ses preuves, il s'agit de l'étude de la jacobienne d'identification au moyen d'une décomposition  $QR$  [Besnard 2001].

#### 4.2.4.1 Vérification de l'identifiabilité avec une décomposition $QR$

Reprenons le système d'équations à résoudre lors de l'identification des paramètres  $\xi$  :  $\Delta F = J_\xi \cdot \Delta \xi$  où  $J_\xi$  est la matrice ( $r \times c$ , avec  $r \gg c$ ) jacobienne de  $F$  par rapport à  $\xi$ . Dans notre cas,  $r$  est égale au nombre de configuration fois le nombre de câbles :  $r = N_C m$  et  $c$  est égal au nombre de paramètres  $a_i$  et  $b_i$  :  $c = 3m + 3m = 6m$ .

Si  $\nu$  est le rang de ce système, alors il peut se réécrire :

$$\Delta F = (J_{\xi_1} \quad J_{\xi_2}) \cdot \begin{pmatrix} \Delta \xi_1 \\ \Delta \xi_2 \end{pmatrix} \quad (4.10)$$

$$J_{\xi_2} = \beta J_{\xi_1} \quad (4.11)$$

où  $J_{\xi_1}$  représente les  $\nu$  colonnes indépendantes de  $J_\xi$  et  $J_{\xi_2}$  représente les  $(c - \nu)$  colonnes dépendantes.  $\beta$  est une matrice ( $\nu \cdot (c - \nu)$ ) constante.

Ainsi l'étalonnage revient à résoudre :

$$\Delta F = J_{\xi_1} \cdot \Delta \xi_\nu \quad (4.12)$$

$$\Delta \xi_\nu = \Delta \xi_1 + \beta \cdot \Delta \xi_2 \quad (4.13)$$

On peut résoudre l'équation (4.12), de rang plein et où le vecteur  $\xi_\nu$  est celui des paramètres identifiables.

Une décomposition  $QR$  de  $J_\xi$  est donnée comme :

$$Q^T \cdot J_\xi = \begin{pmatrix} R \\ 0_{(r-c) \cdot c} \end{pmatrix} \quad (4.14)$$

où  $Q$  est une matrice ( $r \times r$ ) diagonale et  $R$  est une matrice ( $c \times c$ ) triangulaire supérieure.

La théorie (voir section 4.2.2) nous dit que les paramètres non identifiables sont ceux correspondant aux éléments nuls de la diagonale de  $R$ . Ces éléments sont considérés comme numériquement nuls si  $|R_{ii}| < \tau$  avec  $\tau$  un seuil à définir (par exemple en fonction de la précision des mesures  $\varepsilon$  :  $\tau = c.\varepsilon.\max |R_{ii}|$  [Besnard 2001]), et alors  $\xi_i$  n'est pas identifiable.

## 4.3 Expérimentations

### 4.3.1 Mesures

La session d'acquisition des mesures est importante et elle se doit donc d'être préparée. Pour cela nous avons écrit une procédure précise donnée en Annexe A.

L'utilisation d'un appareil de mesure externe, un laser tracker, a été utile. Son utilisation a toutefois nécessité la fabrication de support pour les têtes de mesures sur la plate-forme et sur le bâti du robot (poteaux ou sol) et leur étalonnage.

Le laser tracker (voir figure 4.1), suit une cible (voir figure 4.3) et donne sa position  $(x, y, z)$  dans le repère du laser. Ainsi, en disposant la cible sur des bases disposées de manière fixe sur la plate-forme (voir figure 4.2) ou sur le bâti, le laser fournit la position de ces points dans son repère avec une grande précision et une grande répétabilité. Toutefois, il nécessite plusieurs changement de repère et une bonne connaissance de la position des bases destinées à la cible laser par rapport aux repères du robot.



FIG. 4.1 – Tête du laser tracker

Ce système d'acquisition de mesure va nous permettre d'obtenir les informations nécessaires à l'étalonnage, en mesurant l'état du robot dans un certain nombre de configurations choisies.

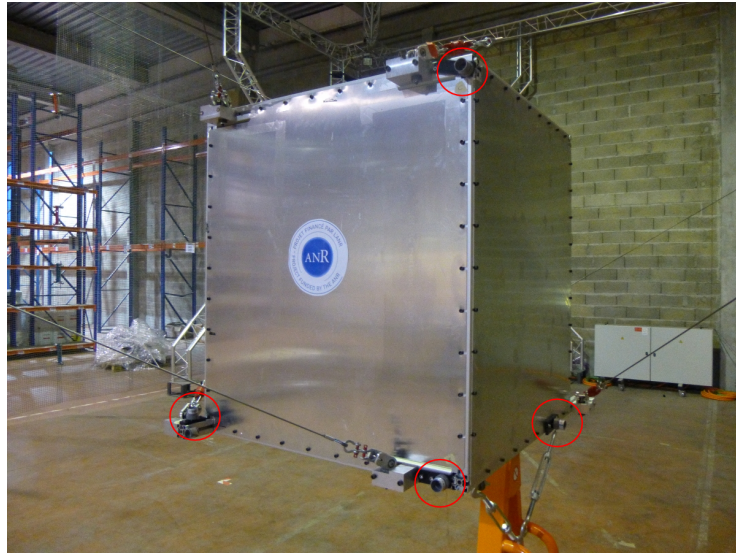


FIG. 4.2 – Bases d'accueil de la cible du laser (entourées en rouge)



FIG. 4.3 – La cible du laser

### 4.3.2 Procédure d'étalonnage du prototype ReelAx8

Le robot reelAx8 est un manipulateur en configuration pleinement contrainte. Il sert de prototype au robot CoGiRo et permet de vérifier et de tester notre approche pour l'étalonnage. Ses caractéristiques sont données en annexe B.

#### 4.3.2.1 Choix des configurations

Afin de réaliser un étalonnage classique, nous avons vu au chapitre 3 que l'hypothèse faite sur les câbles (non pesant et non élastiques) était nécessaire. Nous avons montré que l'erreur engendrée par cette modélisation simplifiée était négligeable pour le robot ReelAx8 au regard de la précision des enrouleurs.

La recherche de poses optimales utilisant l'algorithme 4.2.3.2 a donc été appliquée à tout l'espace de travail du robot ReelAx8, le résultat est visible à la figure 4.4.

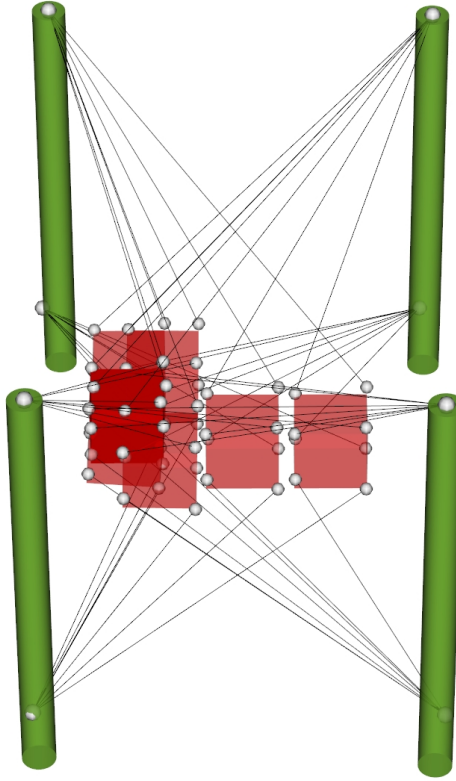


FIG. 4.4 – Poses obtenues pour l'étalonnage du robot ReelAx8

#### 4.3.2.2 Vérification de l'identifiabilité

En ce qui concerne le robot ReelAx8, les faibles orientations possibles font que les paramètres  $a_i$  et  $b_i$  sont faiblement identifiables car ils sont liés par la matrice de rotation de la plate-forme qui reste trop proche de l'identité.

On rappelle l'équation d'étalonnage de la  $k^{eme}$  pose :  $\|p_k + R_k b_i - a_i\| - l_i$ . Si  $R_1 \approx R_k \approx R_{N_C} \approx Id_{3 \times 3}$  alors  $R_k b_i - a_i \approx v_i, \forall k = [1..N_C]$ . Si cette partie de l'équation,  $v_i$ , reste à peu près constante quelles que soient les mesures effectuées, alors seulement un des paramètres pourra être identifié et l'autre devra être fixé.

Ce résultat à été confirmé par la méthode de la décomposition  $QR$ .

Les paramètres  $b_i$ , définissant la plate-forme, sont facilement accessibles à la mesure. On peut donc considérer qu'ils sont connus avec une tolérance suffisamment faible pour la précision finale visée. On cherche donc à étalonner les paramètres  $a_i$ .

Pour l'auto-étalonnage, il faut en plus fixer un repère au bâti, relatif aux points  $a_i$  [Daney 2000], et donc une possibilité est de fixer les paramètres  $a_1^x, a_1^y, a_1^z, a_2^y, a_2^z, a_3^z$ . Ces paramètres ne peuvent alors être identifiés.

#### 4.3.2.3 Identification

Après avoir modélisé les équations d'identification 4.1, choisi les poses d'étalonnage 4.2.3, réalisé les mesures 4.3.1 et vérifié l'identifiabilité des paramètres 4.2, nous pouvons passer à l'étape d'identification.

Nous disposons de 44 mesures pour étalonner et auto-étalonner ce robot. Les poses utilisées pour l'étalonnage du prototype ne proviennent pas de l'algorithme de choix de poses (il n'était pas encore implémenté) mais ont été choisies en fonction de l'expérience. La liste complète est donnée à l'annexe D. Nous en avons écarté 4 qui semblent être aberrantes (erreur de saisie) et nous en avons mis 10 de côté en vue de la validation. Nous étalonnons donc le robot avec 30 mesures. Les tolérances des mesures et des paramètres sont données :

- La précision du laser sur chaque axe (x,y,z) :  $\sigma_{poses} = 2mm$ ,
- La précision des enrouleurs :  $\sigma_{length} = 5mm$ ,
- La qualité de l'estimation des  $A_i$  :  $\sigma_A = 20mm$ ,
- La qualité de l'estimation des  $B_i$  :  $\sigma_B = 10mm$ ,
- La qualité de l'estimation des longueurs perdues :  $\sigma_{\Delta l} = 100mm$ .

**Résultats de l'étalonnage :** Les résultats de l'étalonnage obtenus avec deux méthodes (moindres carrés avec poids et ODR, voir section 1.2.2.1) sont présentés à la figure 4.5. Ils sont donnés sous forme d'erreur résiduelle sur les longueurs de câble. Après seulement 3 itérations, l'algorithme atteint un minimum (avec une erreur moyenne de 3mm pour les moindres carrés pondérés et 1mm pour l'ODR) et après 10 itérations il s'arrête à la précision voulue ( $\Delta F < 10^{-8}$ ).

**Résultats de l'auto-étalonnage :** Les résultats de l'auto-étalonnage obtenus avec deux méthodes (moindres carrés avec poids et  $\chi^2$ , voir section 1.2.2.1) sont présentés à la figure 4.5.

Les variables  $\Upsilon_k = [P, R]_k$  sont initialisées avec un  $MGD$ . Dans notre cas, avec un nombre de câbles supérieur au nombre de degrés de liberté, le  $MGD$  utilisé est sur-contraint et fournit une solution aux moindres carrés. Nous avons remarqué que si l'estimation des paramètres n'est pas suffisamment bonne, le  $MGD$  ne converge pas et le processus d'identification ne peut pas être initialisé. Comme pour l'étalonnage, après 4 itérations l'algorithme atteint un minimum (avec une erreur moyenne de 0.5mm pour les moindres carrés pondérés et 2mm pour le  $\chi^2$ ) et après 10 itérations il s'arrête à la précision voulue ( $\Delta F < 10^{-8}$ ).

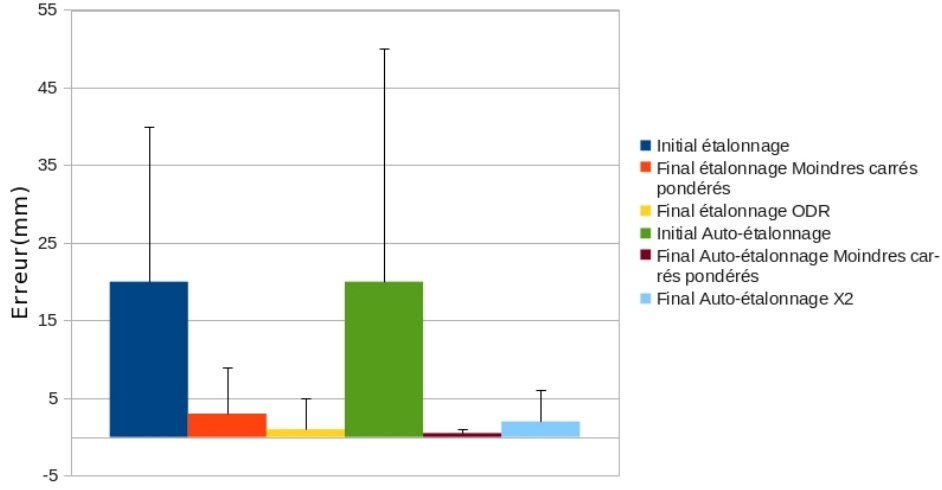


FIG. 4.5 – Moyennes et écarts types des erreurs résiduelles du *MGI* avant et après auto-étalonnage en mm (moyennées sur les 30 poses d'étalonnage et les 8 câbles)

#### 4.3.2.4 Validation

La phase de validation de l'étalonnage consiste à vérifier sur des poses, différentes de celles utilisées pour l'identification, si l'erreur en positionnement et l'erreur en orientation ont été compensées. Elles sont généralement choisies en fonction de la tâche du robot (une phase de pick and place pour un robot dédié au conditionnement par exemple, une trajectoire pour un robot peintre, etc...).

L'erreur entre la pose prédite par le modèle géométrique et la mesure effectuée est calculée avec les paramètres initiaux et avec les paramètres après l'étalonnage. La phase d'étalonnage est validée si l'erreur est diminuée.

Ensuite, il est possible de réaliser une nouvelle expérimentation en mettant à jour les paramètres de la commande et de refaire la même tâche, c'est ce que l'on appelle la compensation. Le comportement du robot doit en être amélioré.

La robot ReelAx8 étant un prototype, il n'est pas vraiment possible de définir des poses de validation en lien avec une tâche. Nous avons donc pris 10 poses au hasard parmi les 40 effectuées afin de réaliser cette validation.

Premièrement, une validation est faite en vérifiant l'amélioration du modèle sur les 10 mesures de validation. Les résultats en terme d'erreur résiduelle sur la longueur des câbles sont présentés à la figure 4.6.

Deuxièmement, nous validons la précision du manipulateur en calculant l'amélioration du positionnement général (l'objectif final de l'étalonnage). Nous mesurons un déplacement entre deux poses de validation, en position  $\Delta P_{mes}$ , et en orientation  $\Delta R_{mes}$  (en angles d'Euler). Ensuite nous calculons le déplacement théorique  $\Delta P_{\lambda}$  et la rotation théorique  $\Delta R_{\lambda}$  avec un *MGD*, où  $\lambda$  signifie la provenance des paramètres utilisés dans le processus (c'est à dire "initiaux", "après l'étalonnage avec les moindres carrés", "après l'étalonnage avec "ODR", "après l'auto-étalonnage avec les moindres carrés", ou "après l'auto-étalonnage avec le  $\chi^2$ "). Les résultats sont donnés



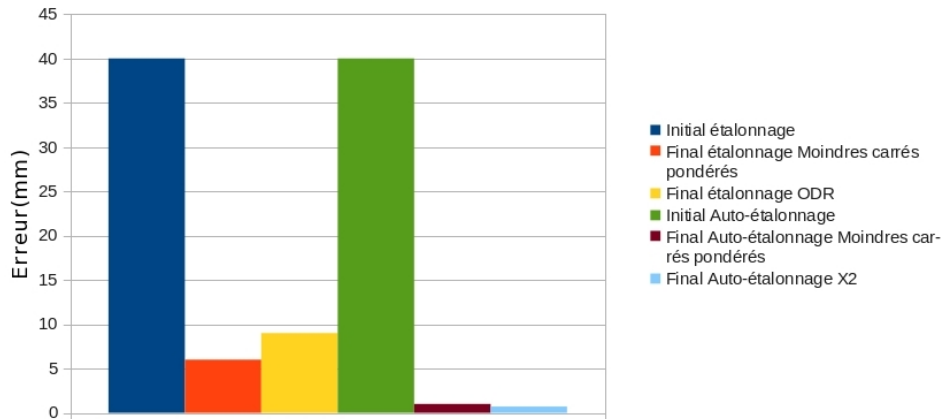


FIG. 4.6 – Moyennes des erreurs résiduelles du *MGI* avant et après auto-étalonnage en mm (moyennées sur les 10 poses de validation et les 8 câbles)

en terme d'erreur relative en pourcentages calculée par  $100 * (\frac{\Delta P_{\lambda} - \Delta P_{mes}}{\Delta P_{mes}})$ . Ils sont présentés sous forme graphique pour un déplacement dans les figures 4.7 pour le positionnement et 4.8 pour l'orientation.

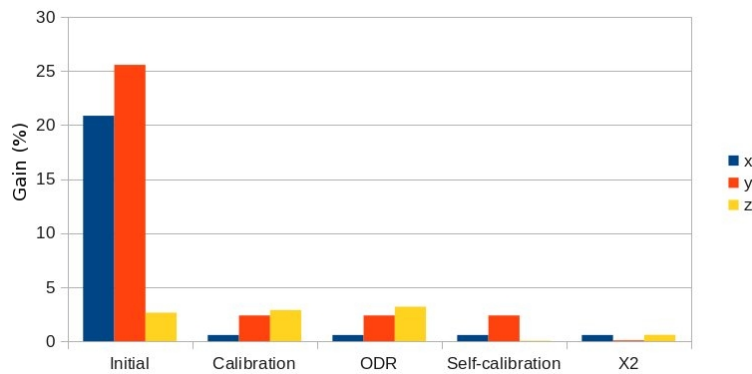


FIG. 4.7 – Erreur en positionnement % (avec les paramètres initiaux et après l'utilisation de 4 méthodes différentes)

On peut voir que l'orientation n'est pas complètement corrigée (excepté pour la méthode du  $\chi^2$ ), probablement parce que les mesures n'ont pas pu être faites dans un grand espace de travail et à des orientations très faibles (moins de 5 degrés).

Toutefois, ces résultats démontrent que l'identification a amélioré la précision du modèle du robot. La phase de compensation n'a pu être possible car le prototype a été reconfiguré pour une autre expérimentation.

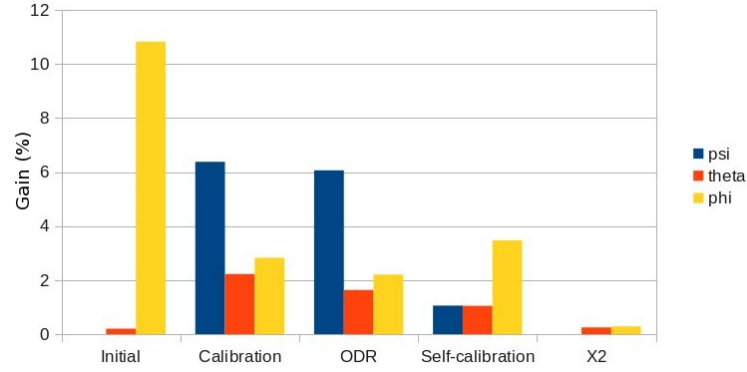


FIG. 4.8 – Erreur en orientation % (avec les paramètres initiaux et après l'utilisation de 4 méthodes différentes)

### 4.3.3 Procédure d'étalonnage du robot CoGiRo

Le robot CoGiRo est le principal manipulateur de cette étude. Il s'agit d'un manipulateur en configuration suspendue. Ses caractéristiques sont données en Annexe C.

#### 4.3.3.1 Choix des configurations

Afin de réaliser un étalonnage classique, on a vu au chapitre 3 que l'hypothèse faite sur les câbles (non pesant et non élastiques) était nécessaire. Cependant, si cette hypothèse est acceptable sur le prototype ReelAx8, on a montré qu'elle entraînait une erreur importante sur la modélisation du robot CoGiRo. Néanmoins, la définition d'un sous espace de travail, dans lequel l'hypothèse est acceptable, a été possible (voir section 3.3.3).

La recherche de poses optimales utilisant l'algorithme 4.2.3.2 a donc été appliquée à un sous espace de travail pour le robot CoGiRo, le résultat est visible à la figure 4.9 et la liste complète est donnée à l'annexe D. Un jeu de pose s'appuyant sur le précepte des poses le plus près des frontières de tout l'espace de travail a aussi été défini, le résultat est donné à la figure 4.10.

#### 4.3.3.2 Vérification de l'identifiabilité

Les résultats pour le robot CoGiRo sont différents de ceux du prototype. En effet, avec des rotations possibles beaucoup plus importantes ( $> 0.4 \text{ radians}$ ), les paramètres  $b_i$  sont identifiables. De même pour l'auto-étalonnage, il faut fixer un repère et donc 6 paramètres du bâti ( $a_1^x, a_1^y, a_1^z, a_2^y, a_2^z, a_3^z$ ) mais également 6 paramètres de la plate-forme ( $b_1^x, b_1^y, b_1^z, b_2^y, b_2^z, b_3^z$ ) ne peuvent être identifiés.

Ces résultats sont confirmés par l'utilisation de la méthode de la décomposition  $QR$ .

#### 4.3.3.3 Identification

Après avoir modélisé les équations d'identification 4.1, choisi les poses d'étalonnage 4.2.3, réalisé les mesures 4.3.1 et vérifié l'identifiabilité des paramètres 4.2, nous pouvons passer à l'étape d'identification. Nous disposons de 5 jeux de mesures :

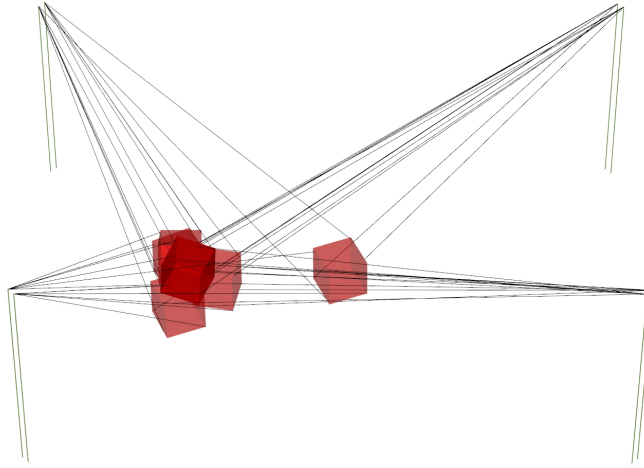


FIG. 4.9 – Poses obtenues pour l'étalonnage du robot CoGiRo dans le sous espace de travail

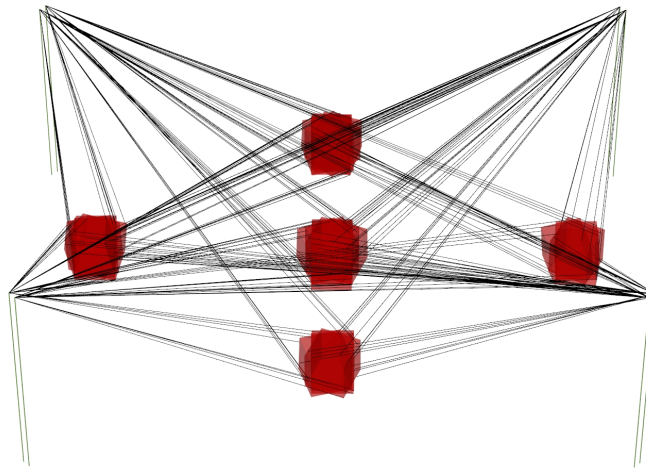


FIG. 4.10 – Poses choisies d'après l'expérience pour l'étalonnage du robot CoGiRo dans tout l'espace de travail

- le jeu  $S_1$  : 12 poses issues de l'algorithme de recherche de poses optimales calculées dans un sous-espace de travail où les câbles peuvent être considérés comme sans masse et sans élasticité (sans entraîner d'erreur de modélisation non négligeable) ;
- le jeu  $S_2$  : 7 poses au centre à des orientations différentes ;
- le jeu  $S_3$  : 23 poses choisies d'après l'expérience sur les bords de l'espace de travail ;
- le jeu  $S_4$  : 6 poses optimales dans l'espace de travail complet du robot ;
- le jeu  $S_5$  : 13 poses de validation obtenues lors d'une tâche de palettisation.

L'expérimentation réalisée sur le prototype ReelAx8 a montré que pour l'étalonnage l'*ODR* était la meilleure méthode car elle prend en compte les erreurs de mesures, et de même pour l'auto-étalonnage le  $\chi^2$  a permis de bons résultats [Sandretto 2012a]. Aux vues de ces résultats, seules ces deux méthodes seront appliquées au robot CoGiRo. Par contre, l'utilisation de différents jeux de mesures est utile pour vérifier certains principes évoqués précédemment.

Les tolérances des mesures et des paramètres sont données :

- La précision du laser :  $\sigma_{poses} = 2mm$ ,
- La précision des enrouleurs :  $\sigma_{length} = 5mm$ ,
- La qualité de l'estimation des  $A_i$  :  $\sigma_A = 100mm$ ,
- La qualité de l'estimation des  $B_i$  :  $\sigma_B = 2mm$ .

**Résultats de l'étalonnage :** L'étalonnage est réalisé avec le critère *ODR* sur les deux jeux de mesures :

- 19 poses situées dans un sous-espace de travail où l'hypothèse du câble est acceptable ( $S_1 \cup S_2$ ).
- 23 poses situées sur les bords de l'espace de travail complet du robot ( $S_3$ ).

Les résultats de l'algorithme sont donnés en terme de moyenne et d'écart type de la valeur du critère pour chaque câble à la figure 4.11 (multipliés par  $10^3$  pour la visibilité). C'est à dire l'erreur entre le modèle et les observations. C'est cette erreur que l'on cherche à minimiser (section 1.2.2.1).

Les valeurs du critère après la phase d'identification utilisant les poses prises dans tout l'espace de travail montrent que l'algorithme de minimisation a réduit l'erreur moyenne d'un facteur 3. On peut tout de suite remarquer que les erreurs initiales sont globalement inférieures (d'un facteur 10) à celles obtenues avec le jeu de mesures réalisé sur tout l'espace de travail. Le modèle est donc plus proche de la réalité dans le sous-espace de travail. Cette observation valide l'approche permettant de définir un sous espace de travail dans lequel l'erreur de modélisation est négligeable. De plus, les valeurs du critère après la phase d'identification utilisant les poses prises dans tout l'espace de travail montrent que l'algorithme de minimisation a réduit l'erreur d'un facteur 4. Il est donc plus aisé pour l'algorithme de trouver des paramètres satisfaisant les observations dans le sous-espace. Ceci confirme également la meilleure approximation du robot par le modèle géométrique simplifié.

**Résultats de l'auto-étalonnage** L'auto-étalonnage est réalisé avec le critère  $\chi^2$  sur les deux jeux de mesures :

- 19 poses situées dans un sous-espace de travail où l'hypothèse du câble est acceptable ( $S_1 \cup S_2$ ).
- 23 poses situées sur les bords de l'espace de travail complet du robot ( $S_3$ ).

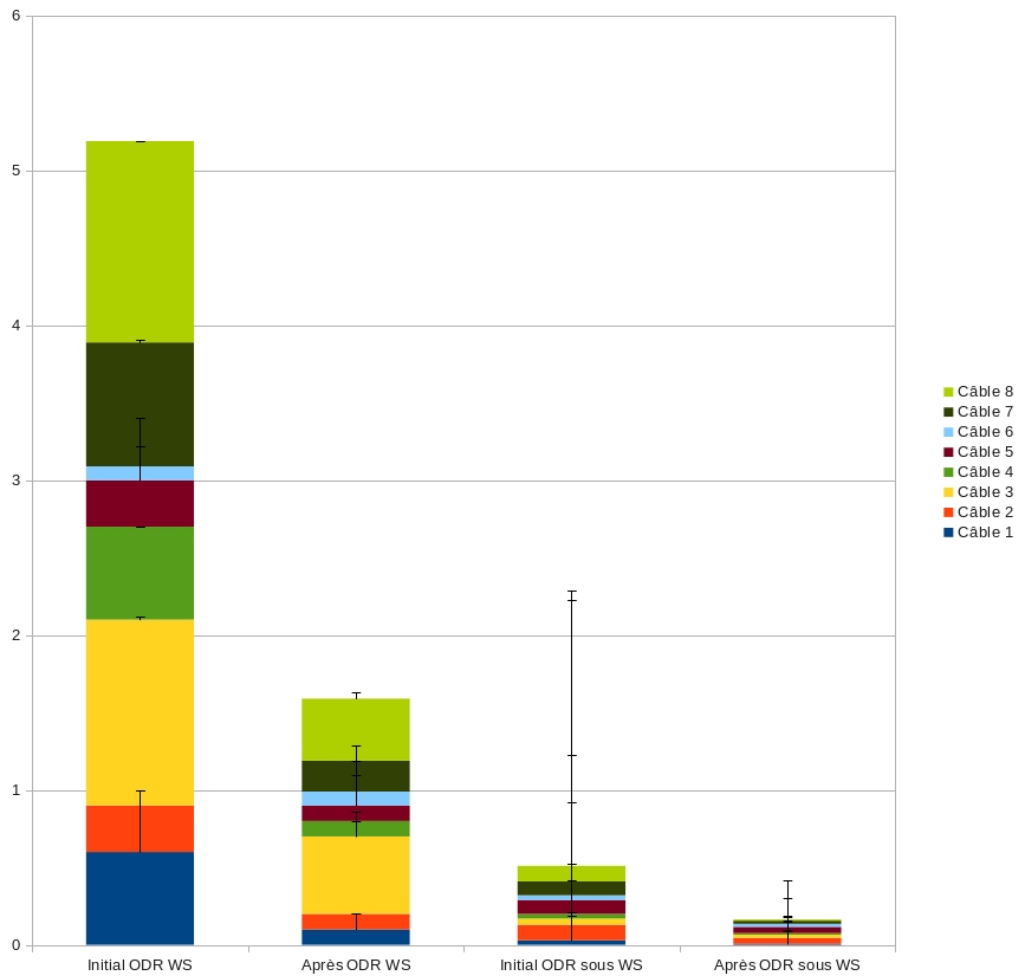


FIG. 4.11 – Moyenne et écart type de la valeur du critère  $ODR$  avant et après étalonnage, avec des poses prises dans tout l'espace de travail  $WS$  ou dans le sous espace, et donnés pour chacun des 8 câbles.

Les résultats de l'algorithme sont donnés en terme de moyenne et d'écart type de la valeur du critère pour chaque câble à la figure 4.12 (multipliés par  $10^3$  pour la visibilité). C'est à dire l'erreur entre le modèle et les observations. C'est cette erreur que l'on cherche à minimiser (section 1.2.2.1).

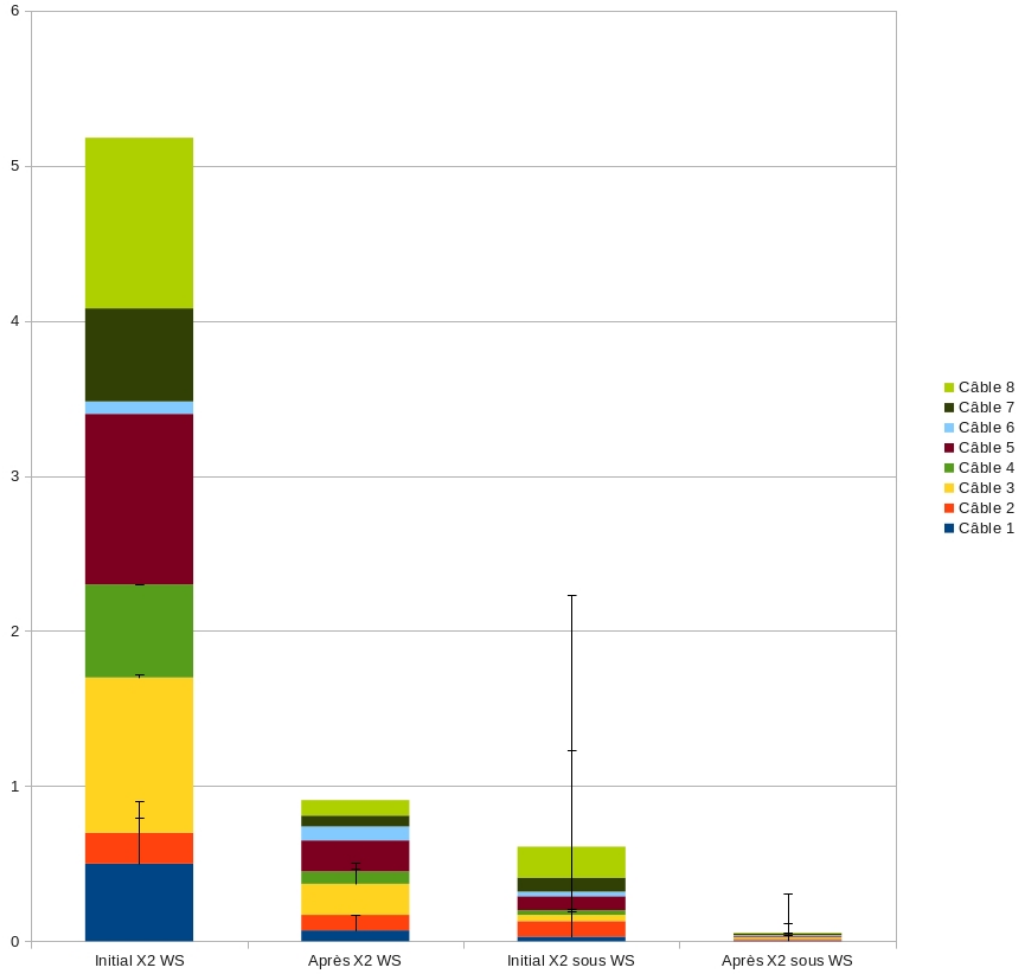


FIG. 4.12 – Moyenne et écart type de la valeur du critère  $\chi^2$  avant et après auto-étalonnage, avec des poses prises dans tout l'espace de travail  $WS$  ou dans le sous espace, et donnés pour chacun des 8 câbles .

Les valeurs du critère après la phase d'identification utilisant les poses prises dans tout l'espace de travail montrent que l'algorithme de minimisation a réduit l'erreur d'un facteur 6. Les valeurs initiales du critère sur les poses prises dans le sous espace de travail implique la même observation qui conduit à confirmer la meilleure qualité du modèle dans le sous-espace de travail (critère 10 fois inférieure). De plus, les valeurs du critère après la phase d'identification utilisant les poses prises dans tout

l'espace de travail montrent que l'algorithme de minimisation a réduit l'erreur d'un facteur 10. Les performances de l'algorithme sont satisfaisantes.

#### 4.3.3.4 Validation

Les poses de validation ont été choisies en fonction de la tâche pour laquelle le robot est conçu : une phase de palettisation. La tâche de palettisation consiste à aller, depuis  $P0$ , chercher une palette en  $P1$ , l'amener en  $P2$  et revenir en  $P0$ . Elle se découpe en 6 poses :

1.  $P0$
2.  $P1$  en bas
3.  $P1$  en haut avec la charge
4.  $P2$  en haut avec la charge
5.  $P2$  en bas
6.  $P0$

Les mesures ont consisté à acquérir la pose de la plate-forme dans les poses 1 – 2 – 3 – 4 – 5 – 6 avec une charge de 35kg, dans les poses 5 – 4 – 3 – 2 – 1 avec une charge de 35kg, et 3 – 4 sans charge. Ce qui nous fournit 13 poses de validation. À noter que la pose  $P1$  est en dehors du sous-espace de travail défini en fonction du modèle de câble.

À la figure 4.13 sont réunies les erreurs du  $MGI$ , c'est à dire la différence entre la longueur de câble déroulée et la distance  $d_i$  en mm, en terme de moyenne et d'écart type sur les 13 poses de validation.

En étalonnant avec les poses prises dans tout l'espace de travail, les erreurs de longueur de câble sont en moyenne plus grandes après étalonnage qu'avant. Ceci signifie que l'étalonnage a détérioré le comportement global du robot. Cependant, en utilisant les poses prises dans le sous espace de travail, les erreurs ont en moyenne légèrement diminué. L'étalonnage a donc amélioré le comportement général du robot. Après l'auto-étalonnage, l'erreur est beaucoup plus grande. Ceci signifie que l'auto-étalonnage a détérioré le comportement global du robot, alors qu'il avait donné un meilleur résultat sur les poses servant à l'identification.

## 4.4 Discussion

L'étalonnage d'un robot est une étape difficile mais nécessaire afin d'améliorer sa précision générale. Les robots à câbles ont un comportement complexe et, lorsqu'ils sont de grande taille, créent un environnement peu confortable pour l'étalonnage. Afin de réaliser un étalonnage et un auto-étalonnage des deux robots étudiés dans ce manuscrit, on a parcouru toutes les phases de l'étalonnage pour les améliorer et les rendre compatibles avec cet environnement difficile.

Ainsi :

- une nouvelle approche de l'auto-étalonnage a été proposée ;
- une jacobienne plus performante est utilisée ;
- des poses optimales ont été calculées ;
- l'identifiabilité a été vérifiée ;
- et des outils d'identification performants ont été employés.

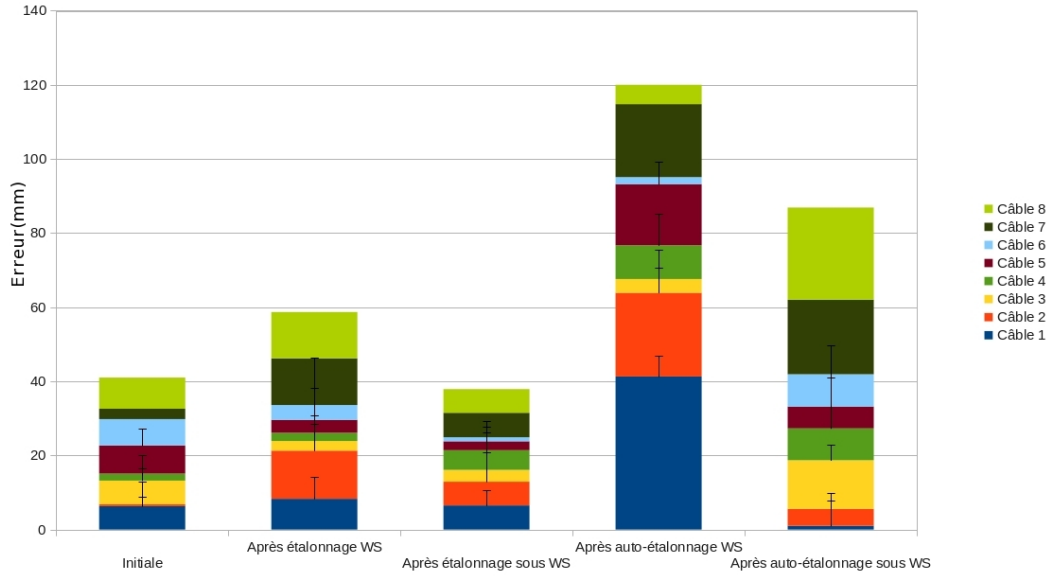


FIG. 4.13 – Moyenne et écart type de l'erreur du *MGI* sur les poses de validation, en mm, avant et après étalonnage et auto-étalonnage et donnés pour chacun des 8 câbles.

Tous ces travaux ont permis un étalonnage et un auto-étalonnage satisfaisant pour le robot ReelAx8, malgré une mauvaise qualité des mesures (conduisant à une mauvaise identifiabilité des paramètres) et une modélisation simplifiée.

Cependant, toutes ces méthodes n'ont pas permis d'améliorer la précision globale du robot CoGiRo.

Néanmoins, la définition d'un sous-espace dans lequel les câbles peuvent être considérés comme des segments droits sans masse, a permis un étalonnage satisfaisant. De plus, l'erreur initiale a montré que le modèle était en effet plus proche de la réalité dans ce sous-espace qu'en dehors, et l'erreur résiduelle a montré que l'identification était plus performante dans ce sous-espace qu'en dehors.

En conclusion, si des outils ont permis de passer certaines difficultés, la nécessité de prendre en compte les erreurs de modélisation des câbles et de considérer la masse de la charge est primordiale.

Au chapitre suivant, nous allons voir une approche permettant de prendre en compte les incertitudes de modèle tout en considérant la possibilité d'embarquer une charge. L'utilisation des intervalles va de plus nous permettre de qualifier notre résultat et ainsi d'avoir une meilleure analyse des résultats obtenus par l'étalonnage du robot CoGiRo.





# Intervalles : étalonnage certifié du robot CoGiRo

---

## Sommaire

<b>5.1</b>	<b>Modélisation du robot CoGiRo</b>	<b>104</b>
<b>5.2</b>	<b>Identifiabilité</b>	<b>106</b>
5.2.1	Nombre de solutions	106
5.2.2	Méthode	106
5.2.3	Résultats expérimentaux	107
<b>5.3</b>	<b>Identification avec le système linéarisé sur-contraint d'intervalles</b>	<b>108</b>
5.3.1	Méthode	108
5.3.2	Résultats expérimentaux	108
5.3.3	Validation de l'étalonnage	109
5.3.4	Discussion	111
<b>5.4</b>	<b>Identification avec une approche programmation par contraintes</b>	<b>112</b>
5.4.1	Différentes approches pour considérer les incertitudes de modélisation du câble	112
5.4.2	Contraction locale des paramètres en fonction d'une pose (le cas contrainte par contrainte)	114
5.4.3	Contraction locale des paramètres en fonction d'un sous système formé de trois poses (le cas bien contraint)	114
5.4.4	Identification avec la <i>q-intersection</i>	116
5.4.5	Résultats expérimentaux	118
5.4.6	Validation de l'étalonnage	123
5.4.7	Discussion	123
<b>5.5</b>	<b>Conclusion</b>	<b>125</b>

---

Les méthodes d'identification de paramètres sont essentiellement basées sur l'approche aux moindres carrés. Cependant, les résultats obtenus avec ces méthodes sont difficilement qualifiables, et l'utilisation de poids pour permettre de considérer les erreurs de mesures ou pour contrôler une éventuelle déviation (voir section 1.2.2.1) entraîne un biais difficilement quantifiable. La validation et la certification du résultat d'un étalonnage sont alors des problèmes complexes et pourtant importants. L'une des solutions à l'étalonnage certifié et qui peut fournir un résultat qualifiable est l'analyse par intervalles. En effet, l'étalonnage de robot utilisant les méthodes de l'analyse par intervalles apporte un résultat numériquement certifié [Daney 2004].

Plus précisément, les méthodes fournissent comme résultat à l'identification un ensemble de solutions qui borne les valeurs possibles des paramètres satisfaisant les

équations d'étalonnage. Si ces méthodes de l'analyse par intervalles ont très peu été appliquées à l'étalonnage de robot, on peut citer [Daney 2004, Baguenard 2005, Daney 2006]. L'identification de paramètres appliquée à d'autres domaines a été étudiée un peu plus largement. Dans le livre [Jaulin 2001], les auteurs proposent une méthode avec une approche ensembliste appelée *SIVIA* pouvant permettre l'identification de paramètres. Des méthodes basées sur l'analyse par intervalles peuvent par exemple permettre l'estimation garantie de paramètres utilisés pour la modélisation en biologie, électronique ou chimie [Walter 2007].

On a vu aux chapitres 1 et 2 que l'étalonnage implique la résolution d'un système sur-contraint qui n'a génériquement pas de solution au sens où elle annulerait toutes les contraintes. Il est néanmoins possible de définir une solution au sens des moindres carrés, c'est à dire qui minimise un critère quadratique (voir section 1.2.2.1). Mais on peut aussi borner les incertitudes sur les paramètres, les mesures ou les modèles avec l'analyse par intervalles. L'approche des intervalles permet de voir le processus d'identification de paramètres non pas comme une solution "au mieux" (aux moindres carrés), mais comme le traitement d'un système sur-contraint d'intervalles.

Ainsi, après une vérification de la présence d'une solution dans la tolérance accordée aux paramètres initiaux, c'est à dire une étude de l'identifiabilité des paramètres au sens intervalles, les méthodes présentées au chapitre 2 seront appliquées à notre problème d'étalonnage du robot CoGiRo.

De plus, on a montré que ces méthodes n'étaient pas robustes aux mesures aberrantes (la solution est influencée par ces mesures). Le système sur-contraint n'ayant pas de solution fiable en présence de ces mesures (voir chapitre 2). Nous proposons donc dans ce chapitre une approche permettant de pallier le problème des mesures aberrantes.

## 5.1 Modélisation du robot CoGiRo

Au chapitre 3, un modèle de câble prenant en compte l'erreur de modélisation maximale entre un câble rigide sans masse et indéformable et un câble réaliste a été défini. L'idée proposée ici est de se baser sur le modèle simplifié du câble et de l'épaissir pour borner les erreurs dues à cette simplification. Ainsi, on peut prendre en compte les erreurs de modélisation des câbles tout en utilisant des contraintes relativement simples.

Cette approche permet d'obtenir un modèle géométrique implicite du robot, en utilisant les intervalles, donné par les  $m$  équations :

$$\|\mathbf{p} + \mathbf{R}\mathbf{b}_i - \mathbf{a}_i\| - (l_i - [\sigma]) = 0, \quad i = [1...m] \quad (5.1)$$

où  $\mathbf{a}_i$  sont les coordonnées du point  $A_i$  dans le repère du robot,  $\mathbf{b}_i$  les coordonnées du point  $B_i$  dans le repère de la plate-forme,  $\mathbf{p}$  et  $\mathbf{R}_{3 \times 3}$  la position et l'orientation de la plate-forme par rapport au repère du robot,  $l_i$  la longueur de câble déroulée et  $[\sigma]$  l'erreur de modélisation.

Le robot CoGiRo est de grande dimension et nous prenons en compte la déformation de la structure en fonction des changements de température et des efforts internes. L'étude *RDM* (la résistance des matériaux) de la structure est disponible et on obtient :

- un déplacement de  $4mm$  vers le bas à cause de la charge (deux câbles par poteau chargés au maximum)
- une variation de  $\pm 2mm$  sur la longueur des poteaux en fonction des changements de température maximaux enregistrés (de  $+10^\circ C$  à  $+35^\circ C$ ).

Nous choisissons de faire porter cette déformation sur les points  $A_i$  qui se trouvent aux extrémités des poteaux. Ainsi une tolérance sur les paramètres  $a_i^z$  due à ces variations est donnée en millimètres par  $\sigma_{A_{rdm}} = [-2.0, 6.0]$  mm. À laquelle s'ajoutent les tolérances des mesures et des paramètres :

- de position  $[\sigma_p]^{x,y,z} = [-2, 2]$  mm,
- d'orientation  $[\sigma_R]^{x,y,z} = [-0.1, 0.1]$  radians,
- de longueur de câble  $[\sigma_l] = [\sigma_d] + [\sigma_L] = [-5, 5] + [-57, 47] = [-62, 52]$  mm,
- des paramètres du bâti  $[\sigma_a]^{x,y} = [\sigma_A] = [-100, 100]$  mm,
- des paramètres du bâti avec la *RDM*  $[\sigma_a]^z = [\sigma_A] + [\sigma_{A_{rdm}}] = [-100, 100] + [-2, 6] = [-102, 106]$  mm,
- des paramètres de la plate-forme  $[\sigma_b] = [-2, 2]$  mm.

Finalement le modèle géométrique prenant en compte les incertitudes de modélisation, des paramètres, des capteurs, de la statique et de la *RDM* est donné par les  $m$  équations :

$$||(\mathbf{p} + [\sigma_p]) + (\mathbf{R} \circ [\sigma_R])(\mathbf{b}_i + [\sigma_b]) - (\mathbf{a}_i + [\sigma_a])|| - (l_i - [\sigma_l]) = 0, \quad i = [1...m] \quad (5.2)$$

où  $(\mathbf{R} \circ [\sigma_R])$  désigne la matrice de rotation  $\mathbf{R}$  composée avec la matrice de rotation issue de la tolérance. Cette composition dépend du formalisme utilisé pour définir  $\mathbf{R}$ , par exemple si on utilise les angles d'Euler  $\psi, \theta, \phi$ , alors  $\mathbf{R}(\psi, \theta, \phi) \circ [\sigma_R] = \mathbf{R}(\psi + [\psi], \theta + [\theta], \phi + [\phi])$ , c'est à dire l'extension naturelle de la matrice de rotation définie avec les angles d'Euler. Ce formalisme entraîne une sur-évaluation de la matrice de rotation qui peut perdre ses propriétés (par exemple l'orthogonalité). Afin de diminuer cela, il est possible d'utiliser le formalisme de Rodrigues (pour calculer la matrice de rotation à partir d'un axe et d'un angle) et la relation  $\mathbf{R}^T \mathbf{R} = \mathbf{Id}$ .

L'objectif de l'étalonnage va être, au moyen de  $N_C$  mesures renseignant  $\mathbf{p}_k$ ,  $\mathbf{R}_k$  et  $l_{i,k}$  de réduire les incertitudes sur les paramètres que l'on souhaite identifier en vérifiant le système d'équations :

$$\begin{aligned} f_{k,i} &= ||(\mathbf{p}_k + [\sigma_p]) + (\mathbf{R}_k \circ [\sigma_R])(\mathbf{b}_i + [\sigma_b]) - (\mathbf{a}_i + [\sigma_a])|| - (l_{i,k} - [\sigma_l]) \\ &= 0, \quad i = [1...m], k = [1...N_C] \end{aligned}$$

Une variable intervalle est ainsi constituée de sa valeur à laquelle on ajoute sa tolérance  $[v] = v + [\sigma_v]$ . Ainsi on peut écrire de manière plus lisible :

$$\begin{aligned} f_{k,i} &= ||[\mathbf{p}_k] + [\mathbf{R}_k][\mathbf{b}_i] - [\mathbf{a}_i]|| - [l_{i,k}] \\ &= 0, \quad i = [1...m], k = [1...N_C] \end{aligned}$$

On l'a vu, la tolérance sur les paramètres  $b_i$  est très faible (ils peuvent être mesurés directement au laser), leur identification n'est donc pas nécessaire. De plus, il est possible d'identifier les paramètres de chaque câble indépendamment. Et ainsi pour chacun des  $m$  câbles on a le système sur-contraint suivant à résoudre :

$$f_k = ||[\mathbf{p}_k] + [\mathbf{R}_k][\mathbf{b}] - [\xi]||^2 - [l_k]^2 \quad (5.3)$$

$$= 0, \quad k = [1 \dots N_C] \quad (5.4)$$

où l'inconnue  $[\xi]$  représente le paramètre  $a_i$  que l'on souhaite identifier.

La jacobienne d'identification est donc, pour chaque câble :

$$J_\xi = \begin{pmatrix} \frac{\partial f_1}{\partial \xi} \\ \vdots \\ \frac{\partial f_{N_C}}{\partial \xi} \end{pmatrix} = - \begin{pmatrix} [\mathbf{p}_1] + [\mathbf{R}_1][\mathbf{b}] - [\xi] \\ \vdots \\ [\mathbf{p}_{N_C}] + [\mathbf{R}_{N_C}][\mathbf{b}] - [\xi] \end{pmatrix} \quad (5.5)$$

## 5.2 Identifiabilité

L'étude de l'identifiabilité des paramètres d'un modèle, en fonction des mesures est un problème complexe à la fois théorique et numérique (voir section 4.2). Elle se réalise classiquement au moyen de l'étude de la matrice jacobienne d'identification [Khalil 1991], afin de vérifier son rang qui doit être égal au nombre de paramètres que l'on cherche à identifier. Cette méthode basée sur une décomposition  $QR$  de la jacobienne permet le cas échéant de déterminer les éventuels couplages entre les paramètres.

Dans le cas d'une modélisation prenant en compte les incertitudes, l'approche d'identification par intervalles est une résolution (ou une contraction) d'un système sur-contraint. L'étude de l'identifiabilité peut donc se ramener à l'analyse de la présence ou du nombre de solutions d'un système sur-contraint [Jaulin 2001, Lagrange 2005].

### 5.2.1 Nombre de solutions

L'unicité de solution aux équations d'identification implique l'identifiabilité des paramètres [Braems 2001]. Dans l'analyse par intervalles, quelques méthodes permettent la vérification de l'unicité (et de l'existence) de solution. On peut citer le test de Krawczyk, l'opérateur de Hansen-Sengupta ou encore celui de Frommer-Lang-Schnurr [Neumaier 1990].

Ces méthodes peuvent être appliquées à l'étude de l'identifiabilité des paramètres du robot.

### 5.2.2 Méthode

Soit  $F(X, \xi)$  le système d'équations d'étalonnage, pour un jeu de mesure donné  $X$ , l'identification consiste à trouver les paramètres  $\xi$  tels que  $F(X, \xi) = 0$ . Pour la lecture, les mesures  $X$  constantes lors de l'identification des paramètres  $\xi$  n'apparaîtront pas et  $F(X, \xi)$  sera noté  $F(\xi)$ . La fonction d'identification  $F$  est supposée continument dérivable en  $\xi$ , et la dérivée s'écrit  $J_\xi$ .

Les méthodes de type Newton par intervalles (voir section 2.1) permettent de calculer une boîte  $N(\hat{\xi}, [\xi])$ . Cette boîte est obtenue en bornant l'ensemble des solutions  $y$  qui satisfont  $F(\hat{\xi}) + J_\xi(\hat{\xi}, [\xi])(y - \hat{\xi}) = 0$  ([Moore 1966, Hansen 2003]). Le terme  $\hat{\xi}$  est un point dans l'intervalle  $[\xi]$ , souvent le point milieu.

Ainsi il est prouvé dans [Neumaier 1990] que si  $N(\hat{\xi}, [\xi])$ , calculé avec une méthode de type élimination de Gauss ou Gauss-Seidel (voir section 1.3.5.1), est inclus dans  $[\xi]$  alors il existe un unique zéro de  $F$  dans  $[\xi]$  :

$$N(\hat{\xi}, [\xi]) \subset [\xi] \Rightarrow \exists! \xi \in [\xi] : F(\xi) = 0 \quad (5.6)$$

Différents tests d'existence et d'unicité peuvent être utilisés, une comparaison de certaines de ces méthodes est réalisée dans [Goldsztein 2007]. L'auteur montre que le test de Hansen-Sengupta est le plus efficace en pratique. Nous employons celui ci dans sa version simplifiée et adaptée à notre problème sur-constraint.

Un algorithme est présenté en 5.2.2 pour étudier l'unicité de la solution. Il est différent des méthodes usuelles dans le sens où il permet de considérer des problèmes avec des paramètres intervalles, comme dans notre cas les mesures. En effet, dû à l'inflation provoquée par ces paramètres,  $N(\hat{\xi}, [\xi])$  peut ne pas être inclus dans  $[\xi]$ .

La méthode vue à la section 2.2.2 est employée pour linéariser le problème et le rendre bien contraint sous la forme  $[A]w = [b]$  avec :

$$[A] = \begin{bmatrix} I & J(\hat{\xi}, [\xi]) \\ J(\hat{\xi}, [\xi])^T & 0 \end{bmatrix}, [w] = \begin{bmatrix} r \\ [\xi] \end{bmatrix}, [b] = \begin{bmatrix} [F(\hat{\xi})] \\ 0 \end{bmatrix}$$

Une méthode pour tester la régularité d'une matrice intervalle est utilisée, elle a été implémentée d'après la méthode *Regularity* de [Rohn 2012].

---

**Algorithme 9** Algorithme pour étudier l'unicité

---

```

Initialiser  $[w] = \begin{bmatrix} [-\infty, \infty] \\ [\xi] \end{bmatrix}$ 
Calcul de  $[A] = \begin{bmatrix} I & J(\hat{\xi}, [\xi]) \\ J(\hat{\xi}, [\xi])^T & 0 \end{bmatrix}$ 
si  $[A]$  est singulière alors
    On ne peut pas conclure à partir de cette matrice (il peut être nécessaire de
    bissecter  $[\xi]$ ).
sinon
    Calcul d'une enveloppe  $[N(\hat{\xi}, [\xi])]$  de l'ensemble des solutions
    si  $[N(\hat{\xi}, [\xi])] \cap [w] = \emptyset$  alors
        Il n'y a pas de solution au problème de l'identification.
    sinon
        si  $[N(\hat{\xi}, [\xi])] \subset [w]$  alors
            Il y a une solution au problème de l'identification et elle est unique (pas
            nécessairement ponctuelle).
        sinon
            On ne peut pas conclure sur le nombre de solutions.

```

---

### 5.2.3 Résultats expérimentaux

L'algorithme 5.2.2 a été appliqué à notre problème d'étalonnage en considérant les systèmes d'équations câbles par câbles (5.3) et les mesures obtenues dans le sous espace de travail ( $S_1 \cup S_2$ ). Nous trouvons que  $[N(\hat{\xi}, [\xi])] \subset [w]$ . Il existe une solution aux  $m$  systèmes dans les boîtes initiales  $[a_i] = a_i^{estim} + [\sigma_a]$ . Ceci signifie, entre autres, que nous avons un jeu de mesures dont aucune n'est aberrante. Les  $m$  jacobiniennes (5.5) sont régulières, l'identification peut donc être réalisée.

### 5.3 Identification avec le système linéarisé sur-contraint d'intervalles

Au chapitre 2, deux méthodes basées sur l'algèbre linéaire ont été présentées afin de fournir une réponse aux caractérisations de la solution d'un système non linéaire et sur-contraint d'intervalles. Il s'agit des méthodes que nous avons nommées : Newton-Bjorck-Hansen-Blik pour déterminer l'enveloppe extérieure et Newton-Bjorck-Beaumont pour obtenir une boîte intérieure.

#### 5.3.1 Méthode

La méthode de résolution de système non linéaire sur-contraint utilisant la notion de reste dans une approche aux moindres carrés, présentée en section 2.2.2, est utilisée afin de proposer un résultat d'étalonnage sous la forme évoquée au paragraphe 2.3. Sous cette forme, le résultat de l'identification sera donné sous la forme d'une enveloppe des valeurs des paramètres garantie contenir toutes les valeurs satisfaisant les contraintes d'étalonnage et d'une boîte intérieure contenant des valeurs satisfaisant l'ensemble des mesures.

Les équations d'étalonnage sont données par le système (5.3) et la jacobienne (5.5).

#### 5.3.2 Résultats expérimentaux

L'identification a été réalisée avec les mesures des  $N_C = 19$  poses sélectionnées dans le sous espace de travail avec la méthode de la section 4.2.3 auxquelles on a ajouté les poses centrales ( $S_1 \cup S_2$ ). Ce jeu de mesures a été vérifié ne contenant aucune mesures aberrantes. Le résultat de l'enveloppe extérieure ( $[x_i]_{out}$ ) est présenté table 5.1. On peut vérifier que  $0 \in f_{k,i}([x_i]_{out}), \forall i = [1...m], \forall k = [1...N_C]$ . On peut également remarquer que la contraction est assez forte, d'un diamètre moyen de 0.2 m (l'incertitude sur les  $a_i$ ), on est passé à un diamètre moyen de 0.03 m, soit une contraction d'un facteur 10. Ainsi la précision garantie sur les paramètres est, après contraction, assez bonne.

	$x$	$y$	$z$
$a_1$	$[-7.1840, -7.1661]$	$[-5.2587, -5.2296]$	$[5.4362, 5.4949]$
$a_2$	$[-7.3267, -7.3052]$	$[-5.1199, -5.0879]$	$[5.4277, 5.5225]$
$a_3$	$[-7.3101, -7.2953]$	$[5.2236, 5.2488]$	$[5.4501, 5.5075]$
$a_4$	$[-7.1740, -7.1479]$	$[5.3520, 5.3947]$	$[5.4254, 5.5547]$
$a_5$	$[7.1770, 7.1876]$	$[5.3361, 5.3595]$	$[5.4692, 5.5111]$
$a_6$	$[7.3165, 7.3305]$	$[5.1888, 5.2236]$	$[5.4633, 5.5412]$
$a_7$	$[7.2968, 7.3065]$	$[-5.1431, -5.1224]$	$[5.4779, 5.5043]$
$a_8$	$[7.1551, 7.1678]$	$[-5.2827, -5.2566]$	$[5.4761, 5.5250]$

TAB. 5.1 – Enveloppe extérieure des paramètres après contraction avec la méthode Newton-Bjorck-Hansen-Blik

Le résultat de la boîte intérieure ( $[x_i]_{in}$ ) est présenté table 5.2. On peut vérifier que  $\forall x \in [x_i]_{in} : 0 \in f_{k,i}(x), \forall i = [1...m], \forall k = [1...N_C]$ .

	$x$	$y$	$z$
$a_1$	$[-7.1750, -7.1750]$	$[-5.2439, -5.2439]$	$[5.4626, 5.4626]$
$a_2$	$[-7.3159, -7.3159]$	$[-5.1030, -5.1030]$	$[5.4722, 5.4723]$
$a_3$	$[-7.3028, -7.3028]$	$[5.2359, 5.2359]$	$[5.4762, 5.4762]$
$a_4$	$[-7.1609, -7.1609]$	$[5.3728, 5.3728]$	$[5.4855, 5.4855]$
$a_5$	$[7.1820, 7.1820]$	$[5.3475, 5.3475]$	$[5.4884, 5.4884]$
$a_6$	$[7.3233, 7.3233]$	$[5.2057, 5.2057]$	$[5.4993, 5.4993]$
$a_7$	$[7.3015, 7.3015]$	$[-5.1325, -5.1325]$	$[5.4891, 5.4891]$
$a_8$	$[7.1612, 7.1613]$	$[-5.2695, -5.2695]$	$[5.4974, 5.4974]$

TAB. 5.2 – Ensemble intérieur tolérable des paramètres avec la méthode Newton-Bjorck-Beaumont

### 5.3.3 Validation de l'étalonnage

La vérification sur les poses de validation consiste à calculer la fonction  $f$  sur les poses de validation avec les paramètres obtenus précédemment. On peut ensuite vérifier différents points :

- $0 \in f_k([x]_{out})$  pour toutes les poses de validation ;
- $\forall x \in [x]_{in} : 0 \in f_k(x)$  pour toutes les poses de validation ;
- La contraction sur les paramètres  $a_i$  a bien réduit l'incertitude sur  $f$ .

Les deux premiers ont été validés. Les résultats sur la réduction de l'incertitude de  $f$  sont exprimés en terme de diamètre moyen des  $f_i$  calculés sur les poses de validation, le résultat est donné sous forme de graphique à la figure 5.1.

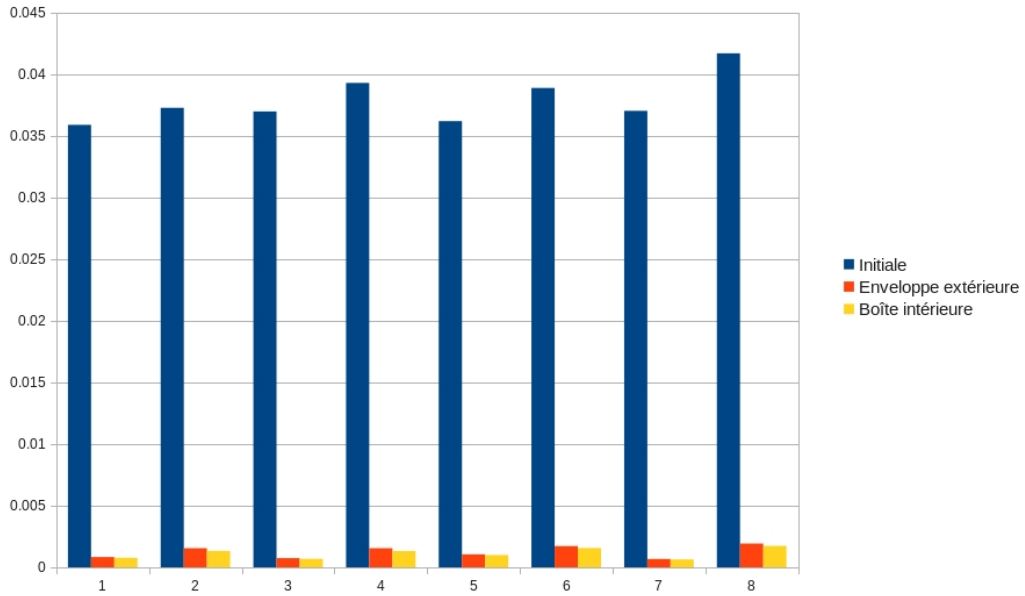


FIG. 5.1 – Résultat sur l'incertitude finale du modèle géométrique pour chacun des huit câbles (en mètres)



Nous disposons donc maintenant d'un jeu de paramètres validant, de manière certifiée, les mesures effectuées tout en considérant les incertitudes de modèles et de mesures.

### 5.3.3.1 Analyse de sensibilité

Dans le but de garantir une précision au robot, une analyse de la sensibilité du robot aux incertitudes de son modèle est possible. Nous rappelons l'équation implicite du robot, pour une pose  $k$  et un câble  $i$  :  $f_{i,k} = \|p_k + R_k b_i - a_i\| - d_{i,k}$

Pour cette analyse on suppose que la distance entre les points d'attache est égale à la longueur de câble déroulée, la différence entre les deux sera considérée comme une incertitude sur la longueur de câble.

En notant les variables de poses  $X = [p, R]$ , les paramètres géométriques  $\xi = [a, b]$  et les mesures proprioceptives  $L = d$ , et en différenciant la fonction  $F = f_{i,k}$  :

$$\begin{aligned} F(X, \xi, L) &= 0 \\ J_X \Delta X + J_\xi \Delta \xi + J_L \Delta L &= 0 \\ \Delta X &= -J_X^{-1}(J_\xi \Delta \xi + J_L \Delta L) \end{aligned}$$

où  $J_* = \frac{\partial F}{\partial *}(X, \xi, L)$ . Les matrices  $J_X$  et  $J_\xi$  sont connues, il s'agit de la matrice jacobienne cinématique (vue au chapitre 4) et de la matrice d'identification (vue au chapitre 4). La jacobienne  $J_L$  étant égale à  $-Id_{m \times 1}$ .

Nous sommes capables de borner les incertitudes sur  $\xi$  (la tolérance  $\pm 10cm$  initialement, puis le rayon de la solution extérieure obtenue 5.1) et les incertitudes sur la différence entre la longueur de câble déroulée et la distance  $d_i$  (la tolérance de l'enrouleur à laquelle on ajoute l'incertitude de modélisation). On a donc les variables intervalles  $[\xi]$  et  $[L]$ .

Ainsi, au pire cas, l'incertitude sur la pose  $X$  notée  $\Delta X$  peut être bornée par :

$$\Delta X_{max} \leq |J_X^{-H}(X, [\xi], [L])| (|J_\xi(X, [\xi], [L])\Delta \xi_{max}| + |J_L(X, [\xi], [L])\Delta L_{max}|)$$

où  $^{-H}$  désigne la hull-inverse calculée d'après la méthode de [Rohn 2012].

Cette borne maximale est calculée aux poses de validation, avant et après étalonnage, et nous gardons les valeurs maximales (au pire cas donc), les résultats sont donnés table 5.3.

	$x$	$y$	$z$	$\psi$	$\theta$	$\phi$
Initialement :	12.6	11.1	24	0.004	0.007	0.022
Après étalonnage :	2.7	2.2	14	0.004	0.007	0.022

TAB. 5.3 – Résultats du calcul de l'incertitude maximale de pose (position en cm et orientation en radians avec le formalisme d'Euler) avant et après étalonnage.

Le robot est donc considéré comme certifié au sens où sa précision est bornée par une erreur en positionnement maximale inférieure aux  $5cm$  exigés sur les axes  $x$  et  $y$ , en considérant toutes les incertitudes du modèle.

### 5.3.4 Discussion

Premièrement, il peut paraître surprenant d'obtenir une boîte englobante de taille non nulle, puisque le point dont on cherche à identifier les coordonnées est par définition ponctuel. Nous avons identifié trois principales causes à cela :

- Il faudrait en théorie prendre une infinité de mesures, ce qui n'est évidemment pas réalisable.
- Sinon il faudrait prendre des mesures permettant une contraction optimale. Ces configurations de mesures peuvent être théoriquement définies par la géométrie, mais ne sont en pratique pas atteignables pour le robot CoGiRo (hors espace de travail).
- Enfin, même en supposant obtenir une infinité de mesures ou celles qui sont optimales pour la contraction, la solution à notre problème d'identification ne sera pas ponctuelle, et nous allons voir pourquoi.

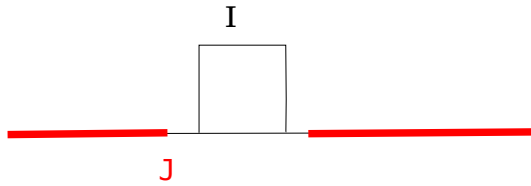
**Solution non ponctuelle :** On peut définir une incertitude comme étant un intervalle borné représentant le support d'une loi de probabilité à support fini. Ainsi la mesure incertaine  $i$  d'une grandeur  $y$  nous donne un support défini par l'intervalle :

$$I_i = \{x : Pr(y = x) > 0\} \quad (5.7)$$

En pratique, cet intervalle n'est pas connu parfaitement, car il faudrait pour cela obtenir la loi de répartition exacte de l'erreur des instruments de mesure. Nous pouvons par contre estimer de manière pessimiste un intervalle  $J_i$  tel que :

$$J_i \subsetneq \bar{I}_i \quad (5.8)$$

On est donc sûr que la probabilité de trouver la valeur de  $y$  est nulle dans l'intervalle  $J_i$ , au prix d'une sur-estimation possible des incertitudes.



Par définition, on peut écrire que  $I_i \subsetneq \bar{J}_i$ . Ainsi, si l'intersection d'une infinité de  $I_i$  donne la valeur quasi-ponctuelle  $y$ , l'intersection des  $\bar{J}_i$  donnera un intervalle qui à l'infini aura un diamètre non nul. En effet,

$$\bigcap I_i \subsetneq \bigcap \bar{J}_i \quad (5.9)$$

On est alors sûr que  $y \notin \bigcup J_i$  et donc que si la grandeur  $y$  existe,  $y \in \bigcap I_i \subsetneq \bigcap \bar{J}_i$ . Il nous faudrait pouvoir calculer l'intersection infinie  $\bigcap I_i$  pour obtenir au mieux  $y$ , mais nous ne disposons que de  $\bigcap \bar{J}_i$  pour en obtenir une enveloppe extérieure. Le diamètre de cette enveloppe nous donne une indication sur la sur-estimation de l'incertitude réalisée en utilisant  $\bar{J}_i$  plutôt que  $I_i$ .

En conclusion, si cette approche a permis d'obtenir de bons résultats en terme de diamètre finale des boîtes contenant les paramètres  $a_i$ , elle ne permet pas de prendre en compte les mesures aberrantes (voir chapitre 2). Le jeu de mesures utilisé a donc dû être rigoureusement vérifié afin d'être certain qu'il ne comporte pas de mesures aberrantes. On a évoqué la possibilité d'obtenir une mesure aberrante si l'incertitude sur un paramètre ou une mesure était mal estimée, pour cela toutes les incertitudes ont dû être sur-estimées par précaution.

De plus, la matrice  $[A]$  en utilisant les équations de Bjorck devient rapidement très grande si on augmente le nombre de mesures ou le nombre de paramètres à identifier.

Afin de pallier le problème de l'absence de solution à cause d'une incertitude mal estimée ou d'une mesure aberrante et également d'utiliser potentiellement plus de mesures, des outils basés sur la programmation par contraintes ont été utilisés pour l'étalonnage du robot CoGiRo.

## 5.4 Identification avec une approche programmation par contraintes

Afin d'améliorer la contraction et donc la précision de l'identification, il est possible d'utiliser plus de mesures. Cependant, l'ajout de mesures implique naturellement l'augmentation du risque d'obtenir une mesure aberrante. Les méthodes utilisées précédemment ne sont pas robustes aux mesures aberrantes et de plus deviennent rapidement très lourdes à utiliser si on ajoute des équations au système considéré. Nous allons définir un nouvel algorithme fondé sur la programmation par contraintes, qui va nous permettre de gérer des systèmes d'équations plus importants (donc plus de mesures) tout en étant robuste aux mesures aberrantes. Pour cela, une intersection capable de relâcher un certain nombre de contraintes est utilisée. Il s'agit de la **q-intersection**, évoquée en section 1.3.6.3.

Dans l'objectif d'améliorer encore la précision de l'identification, nous allons également tenter de diminuer la sur-estimation de l'incertitude de modélisation du câble. Ceci nous permettra de diminuer l'effet décrit au paragraphe 5.3.4.

### 5.4.1 Différentes approches pour considérer les incertitudes de modélisation du câble

Une méthode capable de borner l'erreur ( $\sigma$ ) induite par l'utilisation d'un modèle de câble simplifié (sans masse et sans élasticité) a été introduite à la section 3.3.1.5. Cette approche nous a permis de définir un modèle géométrique implicite "épais", de structure simple, qui est garanti contenir le modèle géométrico-statique du robot :

$$0 \in \|\mathbf{p} + \mathbf{R}\mathbf{b}_i - \mathbf{a}_i\| - ([l_i] - [\sigma_l]), \quad i = [1..m] \quad (5.10)$$

L'incertitude sur la longueur de câble comprend l'incertitude de mesure de l'enroulement  $[\sigma_d]$  et l'erreur de modélisation  $[\sigma_L]$ . Dans la suite de cette section nous parlerons de  $[\sigma]$  pour l'erreur de modélisation du câble  $[\sigma_L]$ . L'erreur  $\sigma$  a, dans un premier temps, été calculée au pire cas, c'est à dire pour la plage maximale des forces qui peuvent être exercées sur la plate-forme du robot (de 40N à 5000N). Mais nous allons voir qu'il est possible d'affiner ce résultat.

## 5.4. Identification avec une approche programmation par contraintes 113

### Au pire cas

L'erreur calculée au pire cas (pc) sur la plage 40-5000N a été donnée comme étant dans l'intervalle  $[\sigma]_{pc} = [-281, 47]$ mm.

### Au pire cas mais adapté à l'étalonnage

Lors de la phase de prise de mesures en vue de l'étalonnage, la masse du chargement n'a pas dépassé 100kg, il est donc possible de restreindre la plage de tension de 40-5000N à 40-1000N. L'erreur due à la modélisation du câble a été donnée comme étant dans l'intervalle  $[\sigma]_{pca} = [-57, 47]$ mm. Nous avons utilisé cet intervalle dans l'étalonnage réalisé à la section précédente.

### Avec l'erreur calculée pour chaque pose d'étalonnage

Nous avons restreint la plage de tension, mais le calcul de l'erreur est réalisé sur tout l'espace de travail, alors que les mesures de pose peuvent nous donner une information plus précise. En effet, les poses ont été mesurées avec un laser, en connaissant la position et l'orientation de la plate-forme, la position des points  $B_i$  dans le repère de référence est connue.

Un changement de repère est nécessaire pour se placer dans le plan du câble centré en  $A_i$  (repère noté  $\Omega_c$ ), plan dans lequel nous pouvons calculer le modèle réaliste d'Irvine [Irvine 1974, Riehl 2011] et ainsi l'erreur commise en le simplifiant (avec la méthode de la section 3.3).

Le vecteur  $\mathbf{u}_i = \overrightarrow{A_i B_i}$  est défini par la géométrie des robots parallèles comme :  
 $\mathbf{u}_i = p + Rb_i - a_i$ . La matrice de rotation  $Q_i = \begin{pmatrix} \cos \gamma & -\sin \gamma & 0 \\ \sin \gamma & \cos \gamma & 0 \\ 0 & 0 & 1 \end{pmatrix}$  avec l'angle  $\gamma = \arctan\left(\frac{u_z}{u_x}\right)$ .

Les coordonnées du point  $B_i$  dans le plan du câble sont alors données par le vecteur  $b_i^{\Omega_c} = |Q_i^{-1} \mathbf{u}_i|$ .

En considérant les incertitudes de mesures et les incertitudes sur les paramètres géométriques  $a_i, b_i$ , ce calcul nous fournit une boîte  $[b_i^{\Omega_c}]$ . Cette boîte remplace l'espace de travail du robot dans le calcul de l'erreur  $[\sigma]$ . Pour  $N_C$  poses et  $m$  câbles, ce processus conduit au calcul de  $2mN_C$  optima  $\sigma_{min}^{i,k}$  et  $\sigma_{max}^{i,k}$ .

Les résultats pour le câble reliant  $A_1$  à  $B_1$  et pour les 10 premières poses d'étalonnage sont donnés à la table 5.4.

Les bornes minimale et maximale calculées pour les 8 câbles et les 48 poses (issues des jeux de mesure  $S_1, S_2, S_3, S_4$ ) sont  $\sigma_{min} = -53$ mm et  $\sigma_{max} = 41$ mm. Nous notons cette incertitude calculée pour chaque pose et chaque câble  $[\sigma_{i,k}]_{cpc}$ .

### En s'appuyant sur la robustesse de la q-intersection

Nous allons utiliser la **q-intersection** (voir section 1.3.6.3), méthode robuste aux mesures aberrantes. Cette robustesse peut être utilisée afin d'utiliser une erreur de modélisation nulle et ainsi de se placer volontairement dans un cas où les incertitudes sont mal estimées.

Dans ce cas là nous allons considérer que l'erreur  $\sigma = 0$ .

$\sigma_{min}$	$\sigma_{max}$
- 20.3931	0.7521
- 22.6900	1.4817
- 23.1533	1.6688
- 20.9929	1.1590
- 31.5952	6.5247
- 19.9187	0.7996
- 32.3839	6.8776
- 38.5759	13.9798
- 44.8950	21.9131
- 39.9349	15.1123

TAB. 5.4 – Résultats du calcul de l’erreur de modélisation en mm pour le câble reliant  $A_1$  à  $B_1$  et pour les 10 premières poses d’étalonnage.

#### 5.4.2 Contraction locale des paramètres en fonction d’une pose (le cas contraint par contrainte)

Pour chaque pose  $k$ , on dispose des mesures de la position ( $\mathbf{p}_k$ ), de l’orientation ( $\mathbf{R}_k$ ) et des longueurs de câble déroulée ( $l_{i,k}$ ). Ces mesures sont données avec une tolérance plus ou moins importante et ainsi on a pu écrire précédemment (voir section 5.1) les équations d’étalonnage géométrique du robot, en prenant en compte les incertitudes de mesure sur les paramètres et sur la modélisation :

$$f_{k,i} = ||[\mathbf{p}_k] + [\mathbf{R}_k][\mathbf{b}_i] - [\mathbf{a}_i]|| - (l_{i,k} - [\sigma_l]) \quad i = [1..m], k = [1..N_C]$$

Nous cherchons ici à réduire l’incertitude sur le paramètre  $a_i$ , c’est à dire à réduire le diamètre de la boîte  $[\mathbf{a}_i]$ . Pour cela, nous utilisons un contracteur.

Pour chaque pose  $k$ , un contracteur de type **HC4-Revise** présenté en introduction (voir section 1.3.6.3) prend en entrée la boîte  $[\mathbf{a}_i]$  et la réduit en fonction d’une seule contrainte  $f_{k,i}$ . Cette étape permet d’obtenir une boîte  $[a_i]_k$  locale à la pose  $k$ . Locale, car en présence d’une mesure aberrante, les boîtes obtenues ne s’intersecteront pas. Cette procédure est appelée pour les  $k = 1..N_C$  et fournit une liste de  $N_C$  boîtes  $[a_i]_{1..N_C}$ , qui seront par la suite intersectées avec une méthode robuste aux mesures aberrantes (la **q-intersection** par exemple).

Nous notons le contracteur défini pour la pose  $k$  et reposant sur **HC4-Revise**  $HC4r(f_{k,i})$ .

Ce contracteur utilise une seule contrainte  $f_{k,i}$  pour contracter  $[a_i]$  sur ses trois dimensions. Il est donc sous-contraint et son efficacité n’est pas suffisante. Nous remplaçons ce contracteur par un contracteur bien contraint basé sur le schéma de Newton.

#### 5.4.3 Contraction locale des paramètres en fonction d’un sous système formé de trois poses (le cas bien contraint)

Par câble  $i$  et par mesure  $k$ , nous disposons d’une seule contrainte  $f_{k,i}$  pour contracter  $[a_i]$  sur 3 dimensions. Cela n’était pas efficace et donc nous avons envisagé des sous-systèmes de 3 contraintes obtenues sur 3 mesures. Nous obtenons alors  $N_C/3$  sous systèmes où chaque mesure n’apparaît qu’une seule fois (afin d’éviter un

## 5.4. Identification avec une approche programmation par contraintes 15

biais). Autrement dit, les  $N_C$  mesures sont partitionnées en sous-systèmes (disjoints) de trois mesures :

$$\begin{cases} f_{s,i} = ||[\mathbf{p}_s] + [\mathbf{R}_s][\mathbf{b}_i] - [\mathbf{a}_i]|| - ([l_{i,s}] - [\sigma_l]) \\ f_{t,i} = ||[\mathbf{p}_t] + [\mathbf{R}_t][\mathbf{b}_i] - [\mathbf{a}_i]|| - ([l_{i,t}] - [\sigma_l]) \\ f_{u,i} = ||[\mathbf{p}_u] + [\mathbf{R}_u][\mathbf{b}_i] - [\mathbf{a}_i]|| - ([l_{i,u}] - [\sigma_l]) \end{cases}$$

avec les indices  $s \neq t \neq u \neq s$ .

Ce système de contraintes, noté  $f_{stu,i}$ , permet d'écrire le contracteur basé sur le schéma de Newton décrit à la section 1.3.6 :  $Newton(f_{stu,i})$ .

Le choix des poses afin d'instancier ce sous système, le choix des indices  $s, t, u$ , a une importance. Nous avons testé trois approches pour les déterminer.

### 5.4.3.1 Choix réalisé dans l'ordre

Dans un premier temps, nous avons pris les contraintes dans l'ordre et ainsi créé l'ensemble des sous systèmes  $\{f_{123,i}, f_{456,i}, f_{...N_C,i}\}$ . Ceci n'est pas la meilleure solution car les mesures sont prises dans un certain ordre, les poses qui se suivent sont souvent relativement proches et conduisent à un mauvais conditionnement du système (voir 4.2).

### 5.4.3.2 Choix visant un conditionnement optimal

Il est possible en théorie de choisir les sous systèmes de manière à les obtenir le mieux conditionnés possible. Nous pourrions par exemple les piocher de façon à avoir un déterminant de la jacobienne du système le plus élevé possible. Si l'approche semble séduisante, en pratique les mesures ne permettent pas l'orthogonalité de tous les sous systèmes. Certains systèmes seront donc assez bien conditionnés mais les derniers seront très mal conditionnés. De plus, cette approche est combinatoire.

### 5.4.3.3 Choix aléatoire

Finalement, pour contrer l'effet du mauvais conditionnement d'un choix réalisé dans l'ordre, tout en essayant d'obtenir quelques sous-systèmes bien conditionnés, nous avons opter pour un tirage aléatoire de sous-systèmes disjoints. Il consiste à mélanger les mesures, afin de conserver le nombre de mesures et l'unicité de chaque mesure puis de les prendre dans l'ordre comme décrit à la section 5.4.3.1.

### 5.4.3.4 Composition de contracteurs

Nous avons vu différents contracteurs à la section 1.3.6 et chacun d'entre eux est plus ou moins efficace suivant le problème à résoudre. Afin d'allier leurs performances, il est intéressant de composer plusieurs contracteurs de nature différente.

Ainsi, le contracteur *Newton* permettant de réduire les boîtes  $[a_i]$  en fonction de trois contraintes (mesures) est composé avec un sous schéma de contraction utilisant le premier contracteur considérant une seule pose et présenté à la section 5.4.2 : *HC4r*.

La programmation par contracteurs (voir section 1.3.6.5 et [Chabert 2009]) nous permet d'écrire simplement le contracteur :

$$[a_i]_{stu} = PtFix(Compo(3BCid(HC4(f_{stu,i}), 2000), Newton(f_{stu,i}))) \quad (5.11)$$

où les contracteurs 3BCid (paramétré pour une découpe en 2000 tranches) et HC4 (HC4r que l'on associe à un algorithme de propagation sur les trois contraintes) ont été présentés à la section 1.3.6.3, et où *PtFix* représente un algorithme de point fixe (il appelle récursivement les contracteurs jusqu'à atteindre une précision donnée).

Ce sous contracteur considérant trois contraintes et donc trois mesures, que l'on note  $Contract_{stu}([a_i])$ , est efficace au point d'être proche de l'optimal (comparaison effectuée avec le calcul de l'optimal sur chaque dimension), pour un calcul très rapide.

*Par exemple : Diamètres obtenus pour  $[a_1]$  lors de la première contraction avec le calcul des optimaux : (0.1923;0.1569;0.2), réalisé en 2.6s.*

*Diamètres obtenus pour  $[a_1]$  lors de la première contraction avec le sous contracteur composé : (0.1925;0.1578;0.2), réalisé en quelques millisecondes.*

#### 5.4.4 Identification avec la q-intersection

Notre procédure d'identification consiste à intersecter les boîtes contenant les paramètres obtenues pour chaque triplet de poses. Ainsi on peut garantir que les paramètres  $a_i$  seront dans l'intersection  $\bigcap_{i=1..N_C} [a_i]_k$ .

Cependant, un problème peut survenir dans cette approche. En pratique, en raison du processus d'acquisition de mesure, une mesure peut être complètement fausse, aberrante, même en considérant la tolérance de l'appareil de mesure. Dans ce cas présent, l'intersection sera vide et l'identification échouera.

Dans le but d'éviter ce phénomène courant, l'intersection classique est remplacée par la **q-intersection** (voir section 1.3.6.3). Cette intersection peut relâcher un certain nombre de contraintes. En effet, le paramètre  $q$  fixe le nombre de boîte à intersecter. Ce paramètre peut être choisi autour de 75 à 90 %, ce qui donne un indice de confiance acceptable pour les mesures. Dans des conditions très difficiles d'acquisition de mesures, une valeur plus faible peut être choisie, cependant, en dessous de 50%, l'étalonnage du robot n'a plus beaucoup de sens (il y aurait plus de mesures aberrantes que de mesures correctes).

Si le résultat de la **q-intersection** est néanmoins vide, il est nécessaire de prendre plus de mesures. Avec cette approche contracteur, plus le nombre de mesures est important, plus le résultat sera affiné. Contrairement à une approche aux moindres carrés où le bruit de mesure et les mesures aberrantes non gérées et en plus grand nombre entraînent un bruit de mesure [Daney 2005] et une déviation des paramètres.

Un exemple est illustré à la figure 5.2. Dans la première image, l'intersection de toutes les boîtes est non vide, une intersection classique fonctionnera. Dans la deuxième image, la présence d'une mesure aberrante impliquera un résultat vide. L'utilisation de la **q-intersection**, avec  $q = 3$ , permettra de trouver la boîte bordurée.

##### 5.4.4.1 Le schéma d'identification

Nous venons de présenter les contracteurs nécessaires à notre procédure d'identification.

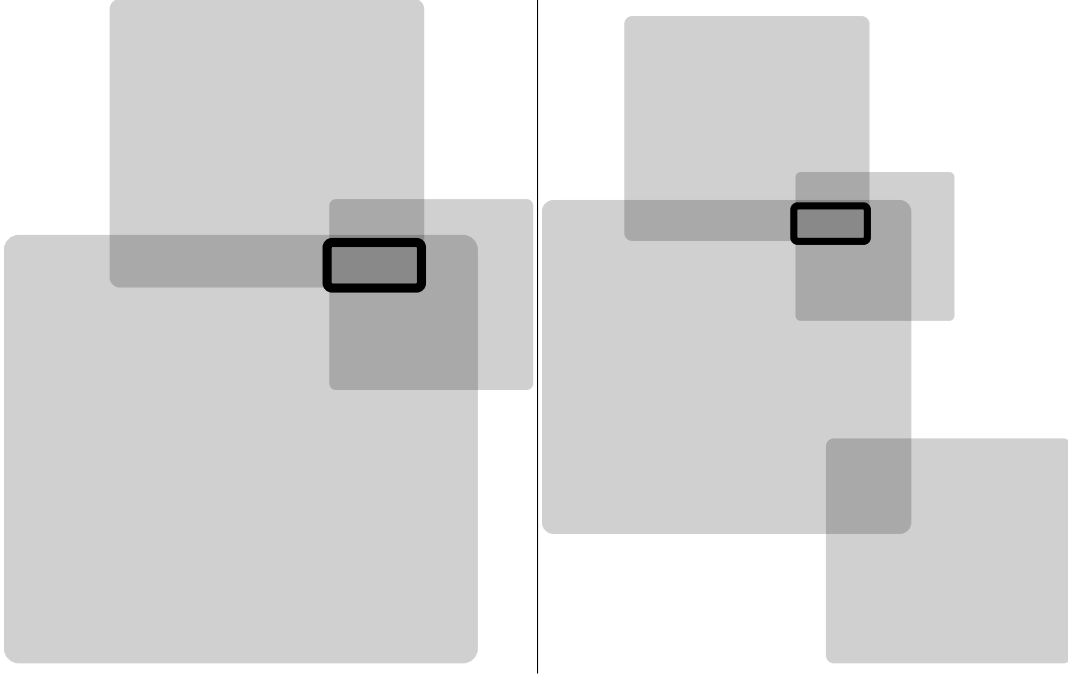


FIG. 5.2 – Intersection de boîtes dans le plan : sans et avec mesure aberrante

On utilise  $Contract_{stu}$  le sous contracteur utilisant les poses  $s, t, u$ , qui fournit les boîtes locales contractées  $[a_i]_{stu}$  tel qu'il a été décrit à la section 5.4.3.4. On note  $Qinter$  le contracteur qui réalise une **q-intersection** des  $[a_i]_{stu}$ . Un contracteur de type point fixe  $PtFix$  appelle le contracteur itérativement jusqu'à ce qu'il ne réduise plus les boîtes passées en argument. C'est lui qui fournit la solution finale de l'étalonnage.

$$[a_i]_{etal} = PtFix(Qinter(q = 90\%, \{Contract_{123}([a_i]), \dots, Contract_{..N_C}([a_i])\}))$$

Ce contracteur global, réduisant la boîte  $[a_i]$  en fonction de  $q$  parmi  $N_C$  mesures prises dans l'ordre de 1 à  $N_C$ , est noté  $Etal_{1,N_C}(q, [a_i])$ .

**Algorithme adaptatif** Ce contracteur, définissant notre schéma d'identification, utilise une approche aléatoire pour piocher des groupes de trois mesures parmi les  $N_C$  mesures disponibles. Ceci peut, en présence de mesures aberrantes, entraîner un biais sur le résultat en le pénalisant si les mesures aberrantes sont réparties dans plusieurs sous-systèmes (encore plus si elles sont dans les sous-systèmes contenant les “meilleures” mesures). Il est donc nécessaire d'essayer plusieurs tirages avant de conclure. De plus, le paramètre  $q$  de la **q-intersection** nécessite d'être le plus élevé possible pour obtenir la solution qui satisfera le plus de mesures effectuées. Il est donc intéressant de commencer à 100% puis de décroître progressivement.

Le contracteur  $Etal_{1,N_C}([a_i])$  est donc appelé dans un algorithme de plus haut niveau présenté en 5.4.4.1.

Le paramètre  $O$  est fixé de telle sorte que la probabilité d'avoir rassemblé au moins deux mesures aberrantes dans le même sous système soit satisfaisante. Nous



**Algorithme 10** Algorithme d'identification

---

```

Mélanger aléatoirement la liste des  $N_C$  mesures
Fixer  $q$  à 100%
Initialiser  $[a_i]$  à la valeur initiale
répéter
  pour  $j = 1 \rightarrow O$  faire
     $[a_i]_{etal} = Etal_{1,N_C}(q, [a_i])$ 
    si  $[a_i]_{etal} \neq \emptyset$  alors
       $[a_i] = [a_i]_{etal}$  (la contraction est conservée)
      Mélanger aléatoirement la liste des  $N_C$  mesures
    Diminuer  $q$  de  $h\%$ 
jusqu'à  $[a_i]$  soit contracté

```

---

avons fixé  $O = 2N_C$ . Ce nombre de tirages élevé permet d'augmenter aussi notre chance d'obtenir un sous système avec un bon conditionnement, ce qui permettra au contracteur *Newton* de réduire de manière quasi optimale la boîte  $[a_i]$ . Le pas  $h$  servant à diminuer progressivement le paramètre  $q$  de la **q-intersection** est fixé à  $h \approx 100/(N_C/3)\%$  afin de s'autoriser à éliminer un sous système à chaque itération.

### 5.4.5 Résultats expérimentaux

L'algorithme 5.4.4.1 a été appliqué à l'étalonnage du robot CoGiRo en utilisant les  $N_C = 48$  mesures prises dans tout l'espace de travail (avec le jeu de mesures  $S_1 \cup S_2 \cup S_3 \cup S_4$ ) en considérant l'erreur de modèle du câble calculée au pire cas, au pire cas adapté à la phase d'étalonnage et au cas par cas (pour chaque pose et chaque câble). Nous obtenons alors trois boîtes pour chaque  $a_i$ , que nous notons  $[a_i]_{pc}$  (pire cas),  $[a_i]_{pca}$  (pire cas adapté),  $[a_i]_{cpc}$  (cas par cas).

L'incertitude sur la modélisation du câble  $[\sigma]$  étant de plus en plus réduite autour de zéro suivant le cas considéré (mais toujours certifiée), nous pouvons écrire :

$$[\sigma]_{pc} \supset [\sigma]_{pca} \supset [\sigma_{i,k}]_{cpc} \quad \forall i = [1..m], \forall k = [1..N_C] \quad (5.12)$$

C'est à dire que nous considérons pour l'étalonnage trois modèles géométriques de moins en moins épaissis par cette incertitude. Le résultat sur  $[a_i]$  devrait donc suivre ce principe et nous devrions obtenir :

$$[a_i]_{pc} \supset [a_i]_{pca} \supset [a_i]_{cpc} \quad (5.13)$$

Cependant, cette supposition n'est vraie que dans un cas idéal où 100% des mesures ont pu être utilisées (**q-intersection** avec  $q = 100\%$ ). Or, il est également probable que le paramètre  $q$  décroisse en fonction de l'épaisseur du modèle :

$$q_{pc} \geq q_{pca} \geq q_{cpc} \quad (5.14)$$

Nous allons distinguer 2 types de résultats. Le cas idéal où l'inclusion est respectée et où 100% des mesures sont satisfaites et celui où une mesure aberrante est détectée au cours du processus de contraction.

#### 5.4.5.1 Inclusion respectée et $q = 100\%$

Les boîtes  $[a_1]$ ,  $[a_2]$ ,  $[a_4]$ ,  $[a_6]$ ,  $[a_7]$  et  $[a_8]$  obtenues avec une **q-intersection** à 100% respectent l'inclusion  $[a_i]_{pc} \supset [a_i]_{pca} \supset [a_i]_{cpc}$ , comme on peut le voir sur la figure 5.3.

#### 5.4.5.2 Mesure aberrante détectée

Les boîtes  $[a_3]$  et  $[a_5]$  obtenues avec une **q-intersection** à 100% respectent l'inclusion  $[a_i]_{pc} \supset [a_i]_{pca}$ , comme on peut le voir sur la figure 5.4. Cependant,  $[a_3]_{pca}$  et  $[a_5]_{pca}$  ont été obtenues avec une **q-intersection** à 90%. Il a donc été nécessaire de relâcher certains sous-systèmes de contraintes (dans notre cas un sur dix-huit).

Dans une étude *a posteriori*, même si le but de l'approche choisie, robuste aux mesures aberrantes, a plutôt vocation à fonctionner comme une méthode "boîte noire", il nous est possible d'identifier la ou les mesures aberrantes nous forçant à relâcher un certain nombre de contraintes. Dans l'exemple de  $[a_3]_{cpc}$ , un des systèmes a pu être enlevé manuellement et le résultat de la contraction ( $[a_3]_{cpc2}$  avec  $q = 100\%$ ) est donné à la figure 5.5. Afin de mieux comprendre le résultat obtenu, toutes les boîtes fournies par le sous contracteur sont présentées à la figure 5.6 pour  $[a_3]_{cpc}$  et à la figure 5.7 pour  $[a_3]_{cpa}$ . On peut ainsi observer que :

- Toutes les boîtes obtenues pour  $[a_3]_{cpa}$  sont plus grandes que celle obtenues pour  $[a_3]_{cpc}$ . Ceci est dû à l'incertitude sur le modèle de câble :  $[\sigma]_{pca} \supset [\sigma_{i,k}]_{cpc} \forall i = [1..m], \forall k = [1..N_C]$  ;
- La solution aberrante trouvée pour  $[a_3]_{cpa}$  est suffisamment sur-évaluée pour s'intersecter avec toutes les autres boîtes ;
- Cette solution aberrante n'en est donc pas une au sens où nous l'avons définie lors du calcul de  $[a_3]_{cpa}$  ;
- Mais elle est aberrante pour le calcul de  $[a_3]_{cpc}$  ;
- La solution  $[a_3]_{cpc}$  est donc correcte, contrairement à  $[a_3]_{cpa}$  qui à cause de la sur-évaluation de l'erreur de modélisation est influencée par une mesure aberrante.

L'inclusion n'est donc pas respectée pour  $[a_3]$  (le principe est le même pour  $[a_5]$ ) car la solution obtenue pour  $[a_3]_{cpa}$  est erronée dans le sens où elle a privilégié une solution aberrante.

*Remarque : En diminuant encore l'incertitude de modélisation des câbles, cette solution aberrante, déjà contenue dans une boîte très petite pour l'incertitude  $[\sigma]_{cpc}$ , devrait sans doute disparaître lors de la sous contraction fournie par le contracteur Newton. Ainsi son élimination ne posera plus de doute.*

#### 5.4.5.3 Sans erreur de modèle

Lors du choix de l'incertitude de modélisation du câble (voir section 5.4.1), qui allait être utilisée pour la définition du modèle géométrique servant à l'identification des paramètres  $a_i$ , nous avons évoqué la possibilité de nous appuyer sur la robustesse de la **q-intersection** pour nous permettre de mal estimer l'incertitude de modèle. De plus, les résultats précédents ont montré qu'une sur-évaluation de l'incertitude de modèle pouvait conduire à la non élimination des solutions aberrantes par la **q-intersection**, ne pas la considérer peut donc conduire à un résultat intéressant.

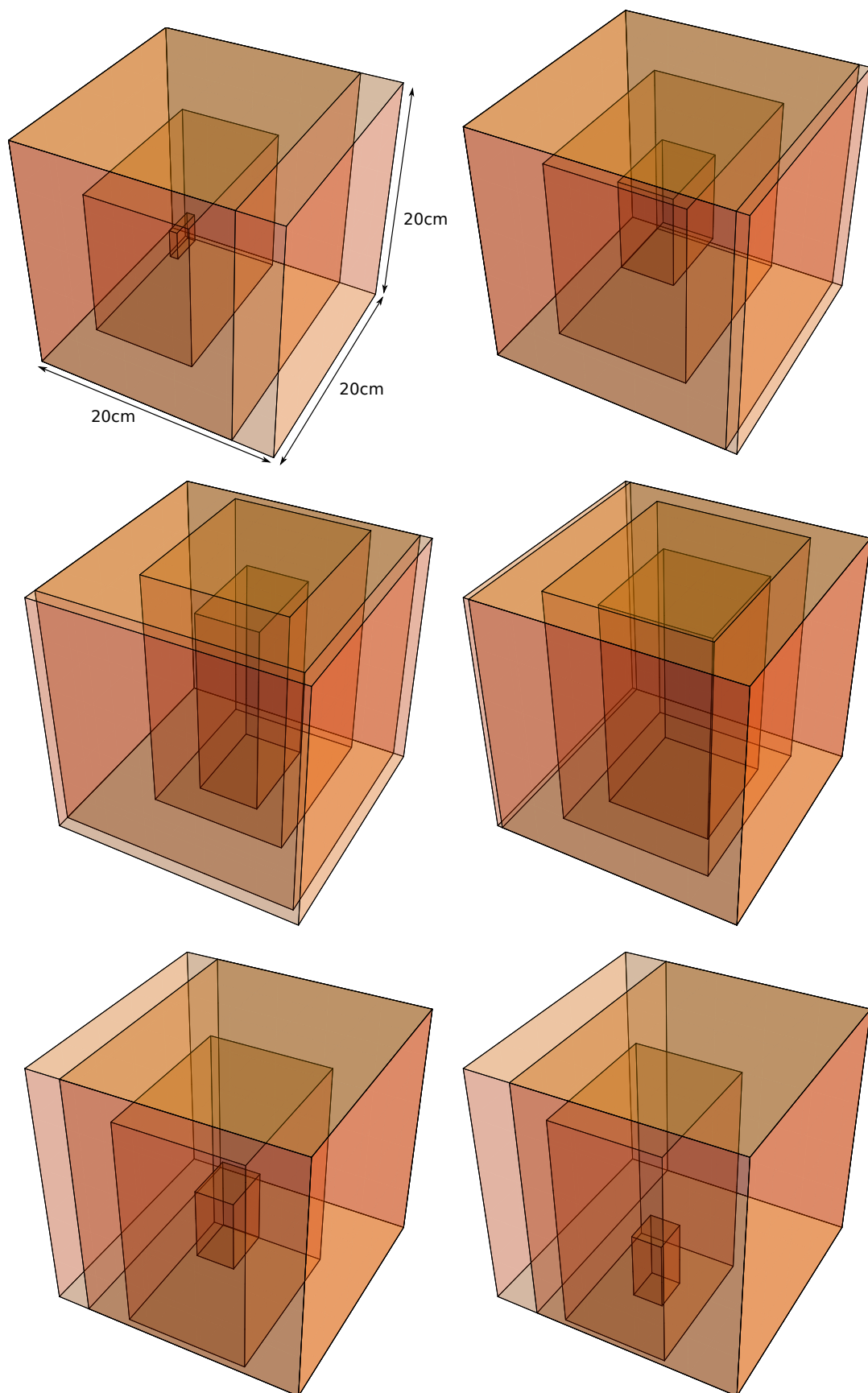


FIG. 5.3 – Résultats (de gauche à droite et de haut en bas) pour  $[a_1]$ ,  $[a_2]$ ,  $[a_4]$ ,  $[a_6]$ ,  $[a_7]$  et  $[a_8]$  : boîte initiale puis du plus grand au plus petit  $[a_i]_{pc}$ ,  $[a_i]_{pca}$ ,  $[a_i]_{cpc}$ .

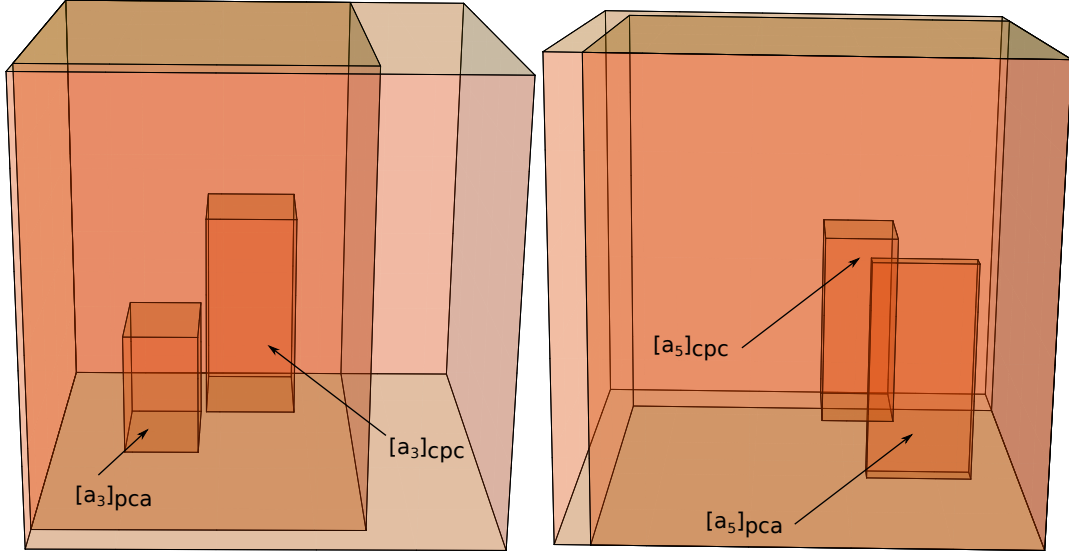


FIG. 5.4 – Résultats pour  $a_3$  et  $a_5$  : boîte initiale et  $[a_i]_{pc}$ ,  $[a_i]_{pca}$ ,  $[a_i]_{cpc}$ .

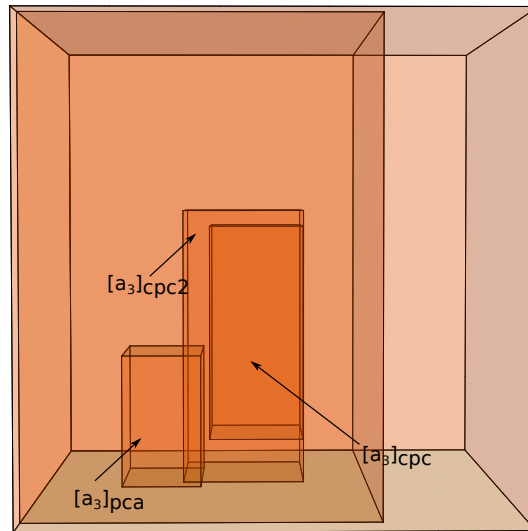


FIG. 5.5 – Résultats pour  $a_3$  avec le système aberrant enlevé manuellement, la solution sans ce système est notée  $[a_3]_{cpc2}$

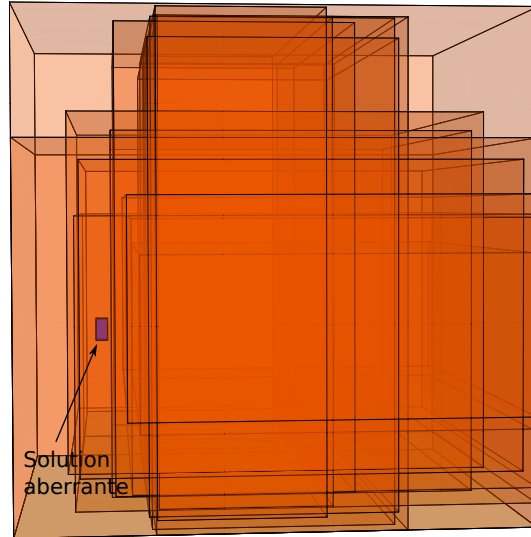


FIG. 5.6 – Résultats des sous contracteurs pour  $[a_3]_{cpc}$ , avec le résultat du sous système aberrant

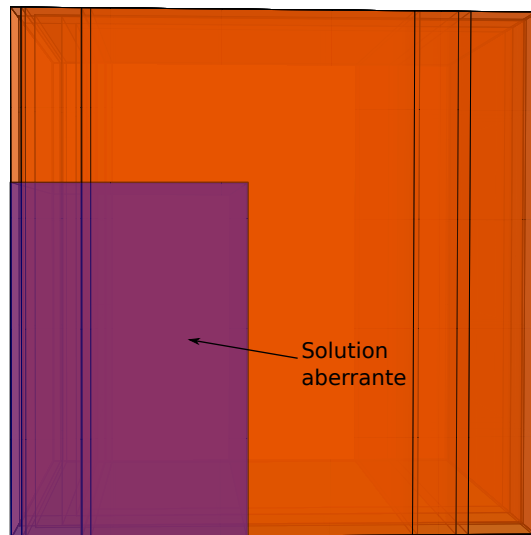


FIG. 5.7 – Résultats des sous contracteurs pour  $[a_3]_{cpa}$ , avec le résultat du sous système aberrant

## 5.4. Identification avec une approche programmation par contraintes 23

Les résultats obtenus avec notre algorithme adaptatif et une incertitude de modélisation nulle sont présentés dans le tableau 5.5.

$i$	$[a_i]_0$	Diamètre moyen	$q$
1	$([-7.177, -7.177]; [-5.282, -5.282]; [5.436, 5.436])$	0.004	90%
2	$([-7.338, -7.303]; [-5.139, -5.100]; [5.432, 5.510])$	0.04	100%
3	$([-7.321, -7.297]; [5.271, 5.292]; [5.367, 5.413])$	0.03	85%
4	$([-7.153, -7.133]; [5.420, 5.444]; [5.385, 5.464])$	0.03	100%
5	$([7.180, 7.223]; [5.362, 5.427]; [5.349, 5.474])$	0.08	75%
6	$([7.288, 7.366]; [5.171, 5.263]; [5.369, 5.602])$	0.13	100%
7	$([7.308, 7.309]; [-5.160, -5.159]; [5.450, 5.451])$	0.0006	80%
8	$([7.150, 7.172]; [-5.344, -5.315]; [5.383, 5.455])$	0.4	90%

TAB. 5.5 – Résultats de l'algorithme d'identification pour  $\sigma = 0$ .

Ces résultats, s'ils ne sont pas tous garantis à 100%, vérifient un pourcentage élevé des mesures effectuées. Ils montrent également l'intérêt de considérer l'erreur de modélisation et d'utiliser une méthode robuste aux mesures aberrantes, sans lesquels ce résultat n'aurait pu être trouvé.

*Remarque : On peut déduire de la faible contraction du paramètre  $a_6$  même avec  $\sigma = 0$  et  $q = 100\%$  que les mesures, ou les incertitudes de mesures, n'ont pas permis de l'identifier avec précision. Ceci nous conduit à penser qu'il y a un travail à mener sur l'identifiabilité (ou la "contractabilité") au sens des intervalles.*

### 5.4.6 Validation de l'étalonnage

Pour certifier le robot étalonné avec la méthode présentée ci-dessus, une approche similaire à celle de la section 5.3.3 utilisant les poses de validation est appliquée. Ainsi, nous avons pu vérifier que la meilleure solution obtenue (la mieux contractée pour chaque  $a_i$ ) satisfaisait **toutes** les équations d'étalonnage instanciées aux poses de validation, même pour les solutions obtenues avec un  $q < 100\%$ .

L'erreur maximale commise sur une pose en fonction des incertitudes présentes (modèle, paramètres géométriques et mesures proprioceptives) est calculée aux poses de validation, avant et après étalonnage. Les résultats sont donnés table 5.6.

	$x$	$y$	$z$	$\psi$	$\theta$	$\phi$
Initialement :	12.6	11.1	24	0.004	0.007	0.022
Après étalonnage :	5.0	3.1	21.1	0.02	0.01	0.03

TAB. 5.6 – Résultats du calcul de l'incertitude maximale de pose (position en cm et orientation en radians avec le formalisme d'Euler) avant et après étalonnage.

### 5.4.7 Discussion

Une nouvelle approche utilisant un principe de composition de contracteurs (appelée la programmation par contracteurs) nous a permis de créer un algorithme d'identification de paramètres robuste aux mesures aberrantes ainsi qu'aux erreurs de modélisation (avec  $\sigma = 0$ ). Différents niveaux de précision ont été utilisés pour borner l'incertitude du modèle de câble, le plus pessimiste, le plus près de la réalité,

ainsi que le plus optimiste. L'algorithme global, adaptatif et reposant sur un principe de simulation par des tirages aléatoires successifs de mesures, est robuste et certifié, et les résultats obtenus sont cohérents. Cependant, les paramètres obtenus restent encore associés à une incertitude importante et ne permettent pas de *garantir* une précision de positionnement inférieure à 5cm. Il est clair dans les résultats obtenus que les faibles contractions le long de l'axe  $z$ , dues à la taille du robot et aux poses ne pouvant être que dans  $1/8$  de l'espace (le parallélépipède engendré par la structure du robot), restent un problème pour ce type d'approche. Des solutions pourraient être de mieux estimer la hauteur des points de sortie des câbles ou de positionner la plateforme en dessous de chaque poteau afin de recueillir des mesures de poses intéressantes.

## 5.5 Conclusion

Dans l'objectif de proposer des méthodes pour étalonner le robot CoGiRo et dont le résultat est certifié et qualifiable, contrairement aux approches des moindres carrés classiques, des algorithmes exploitant l'analyse par intervalles ont été décrits et appliqués. La première approche, décrite au chapitre 2, nous a permis de calculer un sur-ensemble des solutions aux moindres carrés, tout en calculant les résidus des équations d'étalonnage. Ainsi, en vérifiant que ces résidus contiennent zéro, il nous est possible de vérifier qu'une solution satisfaisant pleinement les contraintes existe dans cet ensemble (cette solution n'étant pas forcément ponctuelle). Avec cette approche, nous pouvons donc trouver un sur-ensemble de paramètres qui satisfont le système sur-contraint d'étalonnage. Cependant, la recherche de paramètres vérifiant un système sur-contraint d'équations, comme c'est le cas lors de l'étalonnage, peut conduire à un ensemble vide si on utilise les méthodes classiques des intervalles sans prendre certaines précautions. En effet, nous avons montré dans ce chapitre qu'il était important de borner correctement toutes les incertitudes en présence (mesures, paramètres et modèle) afin d'obtenir une solution à notre problème. De plus, il peut arriver qu'une mesure soit aberrante. Pour être robuste à cette éventualité, qui se produit souvent en pratique, une nouvelle méthode a été développée et testée. Les résultats obtenus avec cette dernière méthode fournissent des informations sur la qualité des mesures effectuées, sur la véracité des tolérances ainsi que sur la "contractabilité" des paramètres. Dans un travail futur, il serait intéressant de coupler ces deux approches (algèbre linéaire et programmation par contraintes) dans un contracteur sur-contraint en remplacement de l'utilisation de sous-systèmes bien contraints.





# Conclusion générale

Dans ce manuscrit de thèse, nous avons présenté les travaux de recherche réalisés sur l'étalonnage des robots parallèles à câbles. Nous proposons ainsi des processus d'identification capables de fournir des solutions dont les caractérisations sont adaptées au contexte. En effet, le robot CoGiRo, principal sujet de nos expérimentations, est un manipulateur :

- de grande échelle mais qui nécessite d'être précis ;
- dont l'architecture semble naturellement robuste aux imprécisions des paramètres géométriques, donc difficilement étalonnable ;
- à câbles, dont les modèles sont complexes ;
- dont la qualité des mesures de tensions dans les câbles n'est pas adaptée à l'étalonnage ;
- dont la mise en œuvre, expérimentalement lourde pour l'étalonnage, n'autorise pas un nombre important de prise de mesures.

## Apports

Le modèle relativement complexe des câbles pesants et élastiques, et l'impossibilité de l'instancier par des informations fiables (les tensions dans les câbles), rend son utilisation problématique. Pour pallier ce problème, nous proposons d'étudier un modèle de câble plus simple mais épaissi afin de prendre en compte les erreurs dues à sa simplification. Une approche intervalles nous a ainsi permis de calculer l'incertitude d'un modèle due à cette simplification. Plusieurs outils permettant la résolution d'un système non linéaire sur-contraint comportant des variables et des paramètres intervalles ont été élaborés pour l'identification des paramètres de ce modèle<sup>1</sup>. L'un d'entre eux a été rendu robuste aux mesures aberrantes et aux erreurs de modélisation.

Nous listons ci-dessous nos principales contributions avec plus de détails.

## Contributions à l'identification de paramètres

### Caractérisation des résultat de l'étalonnage

Une nouvelle réponse a été définie pour un problème d'étalonnage de robot. En s'appuyant sur les intervalles, nous avons pu proposer et caractériser une solution à un problème d'identification de paramètres sous la forme de deux boîtes. Une boîte sur-estimant l'espace solution des paramètres satisfaisant les équations d'étalonnage, en dehors de laquelle nous pouvons garantir qu'aucun paramètre ne sera correct. Cette enveloppe externe a pu être caractérisée par une approche aux moindres carrés (au sens où elle contient l'ensemble des solutions aux moindres carrés) ou par une approche contraction (ce qui implique des solutions qui annulent les contraintes considérées). Une autre boîte a été définie, intérieure celle ci, c'est à dire dans laquelle tous les paramètres géométriques sont corrects, qui a la particularité de

---

<sup>1</sup>Ces outils ont été développés sous Scilab avec la boîte à outils pour les intervalles Int4Sci ou en C++ avec la bibliothèque Ibex.

satisfaire l'ensemble des équations d'étalonnage quelque soient les valeurs possibles des mesures et de leur incertitude. Ce résultat est ainsi beaucoup plus riche pour l'utilisateur, qui peut ensuite exploiter n'importe quel point de la boîte intérieure pour la compensation du manipulateur (c'est à dire les utiliser pour la commande), et se servir de la boîte enveloppante pour garantir la précision ou le comportement du robot pour toute une plage de paramètres.

### Résolution des systèmes non linéaires sur-contraints d'intervalles

Dans l'objectif de fournir les deux boîtes définies comme résultat d'un étalonnage, nous avons dû traiter un système non linéaire sur-contraint d'intervalles. Une approche exploitant la notion de reste (fondée sur les moindres carrés mais avec certaines particularités) a été utilisée pour définir deux algorithmes permettant la résolution globale (dans le sens où toutes les contraintes sont utilisées simultanément) d'un système sur-contraint. L'algorithme fournissant la boîte enveloppante a montré de bonnes performances pour l'étalonnage d'un mécanisme simple. Lors de cette simulation, le deuxième algorithme a permis d'obtenir une boîte intérieure quasi ponctuelle et plus proche de la solution idéale que d'autres méthodes plus classiques (les moindres carrés par exemple). Cependant, en présence de mesures aberrantes, ces méthodes font défaut et conduisent à une solution vide.

## Contributions en robotique des manipulateurs parallèles à câbles

### Modélisation des robots à câbles

Pour étalonner les robots parallèles à câbles, il a tout d'abord été nécessaire de définir un modèle. La principale difficulté repose sur la modélisation des câbles eux mêmes. Plusieurs modèles existent, dont certains sont réalistes mais nécessitent de connaître la tension dans les câbles. Cette information est difficile à obtenir par la résolution d'un modèle géométrico-statique, qui est complexe et encore peu maîtrisé, et actuellement impossible à obtenir par la mesure sur les prototypes étudiés (les capteurs de tension ne sont pas suffisamment fiables). Les intervalles nous ont permis de borner l'incertitude provoquée par l'utilisation d'un modèle de câble simplifié : sans masse ni élasticité. Cette incertitude a permis de définir un modèle géométrique pour le robot, à la fois simple et réaliste, exploitable pour un étalonnage certifié du manipulateur. Nous avons également proposé une méthode pour calculer un sous espace de travail géométrique dans lequel le modèle géométrique classique des robots parallèles était acceptable dans le sens où cette erreur était négligeable au regard d'autres incertitudes.

### Étalonnage des robots parallèles à câbles

Les approches classiques de l'étalonnage et de l'auto-étalonnage des robots parallèles reposant sur les moindres carrés ont été spécialisées à notre problématique. Ainsi les poses d'étalonnage ont été sélectionnées dans le sous espace où nous avons pu limiter l'erreur de modélisation à un seuil acceptable et donc utiliser le modèle géométrique des robots parallèles. De plus, ces poses ont été choisies grâce à un algorithme visant à améliorer l'identifiabilité des paramètres. Enfin, l'étape cruciale de l'identification a été réalisée avec des outils performants. Si l'approche complète

a donné de bons résultats sur le prototype ReelAx8, elle n’a pas permis d’améliorer significativement la précision déjà élevée du robot CoGiRo.

### **Expérimentation de l’étalonnage avec les méthodes intervalles sur le robot CoGiRo**

Pour proposer des méthodes d’étalonnage du robot CoGiRo dont le résultat est certifié et qualifiable, des algorithmes reposant sur les intervalles ont été définis. Afin que ces algorithmes trouvent une solution à un système sur-contraint, nous avons dû dans un premier temps borner de manière sûre (et pessimiste) toutes les incertitudes en présence (mesures, paramètres et modèle). Ainsi le résultat de l’étalonnage, sous la forme des deux boîtes enveloppante et intérieure, a pu être obtenu pour le robot CoGiRo avec une approche aux moindres carrés intervallisés (voir section 5.3).

Dans le but d’utiliser plus de mesures pour affiner le résultat et être robuste aux éventuelles mesures aberrantes, nous avons développé une autre méthode d’identification reposant sur la programmation par contracteurs. L’application de cette méthode a montré son potentiel pour un étalonnage robuste et certifié tout en offrant un certain automatisme (voir section 5.4).

## **Perspectives**

Le principe consistant à épaissir un modèle en fonction de l’incertitude engendrée par des hypothèses simplificatrices peut permettre de répondre à de nombreux problèmes trop complexes ou voire partiellement indéterminés, aussi bien en robotique que dans d’autres domaines. Nos méthodes pour caractériser et obtenir des solutions à l’étalonnage peuvent également être appliquées à tout autre problème d’identification de paramètres.

Plus spécifiquement, nous pensons à quelques travaux futurs en lien direct avec cette thèse que nous listons ci-dessous.

### **Perspectives pour l’identification de paramètres**

#### **Identifiabilité au sens des intervalles**

Au cours des travaux présentés dans cette thèse, il est devenu évident que beaucoup de travaux étaient encore à mener sur l’identification de paramètres avec les intervalles. En réalité, presque tous les résultats connus de la communauté robotique sur la question de l’étalonnage sont à transposer aux intervalles. Nous pensons spécifiquement à l’identifiabilité (nous employons le terme de “contractabilité”) et à la recherche de configurations permettant de l’améliorer. Par exemple, les méthodes actuelles pour le calcul des configurations d’étalonnage pâtissent d’une impossibilité à prendre en compte les incertitudes sur les paramètres et les mesures, ce qui conduit généralement à des résultats approximatifs. L’utilisation des intervalles pour ces questions constituerait donc une approche intéressante.

### Contraction du schéma d'identification

Dans notre schéma d'identification exploitant la programmation par contraintes, nous avons fait le choix de construire un contracteur et de l'utiliser dans un algorithme de point fixe. Cependant, nous avons vu au chapitre 1 que les contracteurs pouvaient également être utilisés dans un schéma de type *Branch & Contract*. En s'inspirant de ce qui a été fait dans [Jaulin 2009b] pour l'extraction robuste de forme dans une image, une perspective pourrait consister à utiliser notre contracteur d'identification dans un schéma de ce type puis de réaliser l'union des enveloppes obtenues. Cette approche plus lourde en calcul pourrait améliorer la qualité de la contraction ou fournir un pavage de l'espace des solutions.

### Perspectives en robotique des manipulateurs parallèles à câbles

#### Modélisation des robots à câbles

Nous avons pu définir un modèle géométrique à la fois simple et prenant en compte la déformation des câbles grâce aux intervalles. Ce modèle intervallisé pourrait conduire, en gérant les incertitudes de manière encore plus fine, à un modèle géométrique direct des robots parallèles à câbles satisfaisant. Ce problème encore ouvert pourrait ainsi bénéficier d'un moyen de réduire l'espace de recherche des solutions, en calculant une enveloppe extérieure aux amas de solutions. De plus, nous savons qu'il est difficile d'avoir six câbles en tension sur les huit du robot CoGiRo, ce qui pourrait conduire à une autre application de la **q-intersection** (éliminer automatiquement les deux câbles qui ne sont pas tendus, au même titre qu'une mesure aberrante).

#### Étalonnage avec la vision

Le projet CoGiRo inclut une commande référencée vision. Ce dispositif n'était pas opérationnel assez tôt dans l'avancement des travaux présentés dans ce manuscrit. Cependant, déjà quelques approches ont été initiées pour identifier les paramètres géométriques du robot et du système de vision au moyen exclusif des données obtenues par le réseau de caméras. Certaines études sont prometteuses et pourraient être poursuivies.

# Bibliographie

- [Aberth 1997] O. Aberth. *The solution of linear interval equations by a linear programming method*. Linear algebra and its applications, 1997. (Cité en pages 33 et 35.)
- [Agahi 2010] M. Agahi et L. Notash. *Trajectory Planning and Impact Reduction in Wire-Actuated Parallel Manipulators With Elastic Wires*. 2010. (Cité en pages 17 et 69.)
- [Albus 1993] J. Albus, R. Bostelman et N. Dagalakis. *The NIST Robocrane*. Journal of Robotics Systems, vol. 10, pages 709–724, 1993. (Cité en page 11.)
- [Araya 2008] I. Araya, B. Neveu et G. Trombettoni. *Exploiting Common Subexpressions in Numerical CSPs*. In Proc. CP, Constraint Programming, LNCS 5202, pages 342–357. Springer, LNCS, 2008. (Cité en page 71.)
- [Araya 2010] I. Araya, G. Trombettoni et B. Neveu. *Exploiting Monotonicity in Interval Constraint Propagation*. In Proc. AAAI, pages 9–14, 2010. (Cité en page 37.)
- [Araya 2012] I. Araya, G. Trombettoni et B. Neveu. *A contractor based on convex interval taylor*. Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems, pages 1–16, 2012. (Non cité.)
- [Baguenard 2005] X. Baguenard. *Propagation de contraintes sur les intervalles, application à l'étalonnage des robots*. PhD thesis, Université d'Angers, 2005. (Cité en page 104.)
- [Barrette 2000] G. Barrette. *Analyse des mécanismes parallèles actionnés par câbles*. Master's thesis, Université Laval, Québec, 2000. (Cité en page 65.)
- [Barrette 2005] C. Barrette G. and Gosselin. *Determination of the dynamic workspace of cable-driven planar parallel mechanisms*. Journal of Mechanical Design, vol. 127, pages 242–248, 2005. (Cité en page 17.)
- [Beaumont 1999] O. Beaumont. *Algorithmique pour les intervalles : Comment obtenir un résultat sûr quand les données sont incertaines*. PhD thesis, Université de Rennes, 1999. (Cité en pages 35 et 48.)
- [Behzadipour 2005] S. Behzadipour. *High-speed Cable-based Robots with Translational Motion*. PhD thesis, University of Waterloo, Canada, 2005. (Cité en page 15.)
- [Behzadipour 2006] A. Behzadipour S. and Khajepour. *Stiffness of Cable-based Parallel Manipulators with Application to the Stability Analysis*. ASME Journal of Mechanical Design, vol. 128, pages 303–310, 2006. (Cité en page 16.)
- [Benhamou 1999] F. Benhamou, F. Goualard, L. Granvilliers et J.-F. Puget. *Revising Hull and Box Consistency*. In Proc. ICLP, pages 230–244, 1999. (Cité en page 37.)
- [Bennet 1991] D.J. Bennet et J.M. Hollerbach. *Autonomous calibration of single-loop closed kinematic chains formed by manipulators with passive endpoint constraints*. 1991. (Cité en page 22.)

- [Bentbib 2002] A. H Bentbib. *Solving the full rank interval least squares problem*. Appl. Numer. Math., vol. 41, pages 283–294, may 2002. (Cité en page 43.)
- [Berti 2013] A. Berti, J.-P. Merlet et M. Carricato. *Solving the Direct Geometrico-Static Problem of 3-3 Cable-Driven Parallel Robots by Interval Analysis : Preliminary Results*. In Tobias Bruckmann et Andreas Pott, éditeurs, Cable-Driven Parallel Robots, volume 12 of *Mechanisms and Machine Science*, pages 251–268. Springer Berlin Heidelberg, 2013. (Cité en page 17.)
- [Besnard 1999] S. Besnard et W. Khalil. *Calibration of Parallel Robots with Two Inclinometers*. In Robotics and Automation, volume 3, pages 1758–1763, 1999. (Cité en page 21.)
- [Besnard 2000] S. Besnard. *Etalonnage geometrique des robots séries et paralleles*. PhD thesis, Ecole Centrale de Nantes, 2000. (Cité en page 20.)
- [Besnard 2001] S. Besnard et W. Khalil. *Identifiable Parameters for Parallel Robots Kinematic Calibration*. In International Conference on Robotics and Automation, volume 3, pages 2859–4729, Seoul, Korea, may 2001. (Cité en pages 27, 85, 88 et 89.)
- [Björck 1967] A. Björck et G. H. Golub. *Algol programming : Iterative Refinement of Linear Least Squares Solutions by Householder Transformation*. j-BIT, vol. 7, no. 4, pages 322–337, dec 1967. (Cité en page 44.)
- [Boggs 1987] P. T. Boggs, R. H. Byrd et R. B. Schnabel. *A Stable and Efficient Algorithm for Nonlinear Orthogonal Distance Regression*. SIAM Journal on Scientific and Statistical Computing, vol. 8, no. 6, pages 1052–1078, 1987. (Cité en page 24.)
- [Bonev 2003] L. Bonev. *The true origins of parallel robots*. 2003. (Cité en page 6.)
- [Borgstrom 2009a] P. H. Borgstrom. *Novel cable-driven robotic platforms and algorithms for environmental sensing applications*. PhD thesis, University of California, Los Angeles, 2009. (Cité en page 15.)
- [Borgstrom 2009b] P.H. Borgstrom, B.L. Jordan, B.J. Borgstrom, M.J. Stealey, G.S. Sukhatme, M.A. Batalin et W.J. Kaiser. *NIMS-PL : A Cable-Driven Robot With Self-Calibration Capabilities*. Robotics, IEEE Transactions on, vol. 25, no. 5, pages 1005 –1015, oct. 2009. (Cité en pages 11, 22 et 62.)
- [Borm 1991] J.-H. Borm et C.-H. Menq. *Determination of optimal measurement configurations for robot calibration based on observability measure*. Int. J. Rob. Res., vol. 10, pages 51–63, jan 1991. (Cité en page 87.)
- [Bosscher 2007] P. Bosscher et al. *Cable-suspended robotic contour crafting system*. Automation in Construction, vol. 17, no. 1, pages 45 – 55, 2007. (Non cité.)
- [Bouchard 2008] S. Bouchard. *Geometrie des robots paralleles entraines par des cables*. PhD thesis, Universite Laval, Quebec, 2008. (Cité en pages 15, 62 et 69.)
- [Braems 2001] I. Braems, L. Jaulin, M. Kieffer et E. Walter. *Guaranteed numerical alternatives to structural identifiability testing*. In Decision and Control, 2001. Proceedings of the 40th IEEE Conference on, volume 4, pages 3122 –3127 vol.4, 2001. (Cité en page 106.)
- [Brethé 2011] J.-F. Brethé. *Modélisation de la précision micrométrique des robots manipulateurs pilotés à l'aide d'informations extéroceptives*. 20ème Congrès

- Français de Mécanique, 28 août/2 sept. 2011-25044 Besançon, France (FR), 2011. (Non cité.)
- [Brethé 2012] J.-F. Brethé. *Caractérisation, Dimensionnement et Commande des robots manipulateurs industriels pour une meilleure précision (HDR)*. PhD thesis, UFR de Science du Havre, 2012. (Cité en page 19.)
- [Bruckmann 2008a] T. Bruckmann, L. Mikelsons, M. Hiller et D. Schramm. *Continuous Workspace Analysis, Synthesis and Optimization of Wire Robots*. ASME Conference Proceedings, vol. 2008, no. 43260, pages 59–66, 2008. (Cité en page 66.)
- [Bruckmann 2008b] T. Bruckmann, L. Mikelsons, M. Hiller et D. Schramm. *Continuous Workspace Analysis, Synthesis and Optimization of Wire Robots*. ASME Conference Proceedings, vol. 2008, no. 43260, pages 59–66, 2008. (Cité en page 17.)
- [Carricato 2012a] M. Carricato, G. Abbasnejad et D. Walter. *Inverse Geometrico-Static Analysis of Under-Constrained Cable-Driven Parallel Robots with Four Cables*. In Jadran Lenarcic et Manfred Husty, éditeurs, Latest Advances in Robot Kinematics, pages 365–372. Springer Netherlands, 2012. (Cité en pages 16 et 18.)
- [Carricato 2012b] M. Carricato et J.-P. Merlet. *Stability Analysis of Underconstrained Cable-Driven Parallel Robots*. IEEE Transactions on Robotics, pages 1 – 9, 2012. (Cité en pages 11, 16, 61 et 65.)
- [Chabert 2009] G. Chabert et L. Jaulin. *Contractor programming*. Artificial Intelligence, 2009. (Cité en pages 39 et 115.)
- [Chablat 2004] D. Chablat, Ph. Wenger, F. Majou et J. P. Merlet. *An Interval Analysis Based Study for the Design and the Comparison of Three-Degrees-of-Freedom Parallel Kinematic Machines*. The International Journal of Robotics Research, vol. 23, no. 6, pages 615–624, 2004. (Cité en page 28.)
- [Chebyshev 1874] P. Chebyshev. *Sur les valeurs limites des integrales*. Journal des Mathématiques Pures et Appliquées, 1874. (Cité en pages 28 et 43.)
- [Corbel 2008] D. Corbel. *Contribution à l'amélioration de la précision des robots parallèles*. PhD thesis, Université Montpellier II, 2008. (Cité en page 20.)
- [Dallej 2011] T. Dallej, M. Gouttefarde, N. Andreff, M. Michelin et P. Martinet. *Towards vision-based control of cable-driven parallel robots*. In Intelligent Robots and Systems (IROS), pages 2855–2860, Sept. 2011. (Cité en page 12.)
- [Dallej 2012] T. Dallej, M. Gouttefarde, N. Andreff, R. Dahmouche et P. Martinet. *Vision-based modeling and control of Large-Dimension Cable-Driven Parallel Robots*. In IEEE, éditeur, Conference on Intelligent Robots and Systems., pages 1581–1586, Vilamoura, Algarve, Portugal, 2012. (Cité en pages 12, 17 et 80.)
- [Daney 1999] D. Daney. *Self calibration of Gough platform using leg mobility constraints*. In in Proc. 10th World Congress Theory of Machine and Mechanisms, Oulu, Finland, pages 104 – 109, 1999. (Cité en page 22.)
- [Daney 2000] D. Daney. *Etalonnage géométrique des robots parallèles*. PhD thesis, Nice - Sophia Antipolis, 2000. (Cité en pages 20, 21, 22, 23, 82 et 92.)



- [Daney 2002] D. Daney. *Optimal Measurement Configurations for Gough Platform Calibration*. In ICRA, Washington, US, volume 4, pages 147 – 152, 2002. (Cit  en page 86.)
- [Daney 2004] D. Daney, Y. Papegay et A. Neumaier. *Interval methods for certification of the kinematic calibration of parallel robots*. In Robotics and Automation, 2004. Proceedings. ICRA '04. 2004 IEEE International Conference on, volume 2, pages 1913 – 1918 Vol.2, 26-may 1, 2004. (Cit  en pages 103 et 104.)
- [Daney 2005] D. Daney, Y. Papegay et B. Madeline. *Choosing Measurement Poses for Robot Calibration with the Local Convergence Method and Tabu Search*. International Journal of Robotics Research, vol. 24, no. 6, pages 501–518, 2005. (Cit  en pages 86, 87 et 116.)
- [Daney 2006] D. Daney, N. Andreff, G. Chabert et Y. Papegay. *Interval method for calibration of parallel robots : Vision based experiments*. Mechanism and Machine Theory, vol. 41, pages 929–944, 2006. (Cit  en pages 28, 43 et 104.)
- [Deblaise 2006] D. Deblaise. *Contribution   la mod lisation et   l talonnage  lasto-g om trique des manipulateurs   structure parall le*. PhD thesis, Institut National des Sciences Appliqu es, Rennes, 2006. (Cit  en page 20.)
- [Deif 1988] G. Zielke Deif A. *Sensitivity Analysis in Linear Systems*. ZAMM - Journal of Applied Mathematics and Mechanics / Zeitschrift f r Angewandte Mathematik und Mechanik, vol. 68, no. 8, pages 346–346, 1988. (Cit  en pages 43 et 44.)
- [Deif 1991] A. Deif. *The Interval Eigenvalue Problem*. ZAMM - Journal of Applied Mathematics and Mechanics / Zeitschrift f r Angewandte Mathematik und Mechanik, vol. 71, no. 1, pages 61–64, 1991. (Non cit .)
- [Dekker 2006] R. Dekker, A. Khajepour et S. Behzadipour. *Design and Testing of an Ultra High-Speed Cable Robot*. Journal of Robotics and Automation, vol. 21, pages 25–34, 2006. (Cit  en page 15.)
- [Ding 2005] F. Ding et T. Chen. *Hierarchical least squares identification methods for multivariable systems*. Automatic Control, IEEE Transactions on, vol. 50, no. 3, pages 397 – 402, march 2005. (Non cit .)
- [Driels 1990] M. R. Driels et U. S. Pathre. *Significance of observation strategy on the design of robot calibration experiments*. Journal of Robotic Systems, vol. 7, no. 2, pages 197–223, 1990. (Cit  en page 86.)
- [Everett 1987] L. J. Everett, M. Driels et B. W. Mooring. *Kinematic modelling for robot calibration*. In Robotics and Automation, volume 4, pages 183 – 189, mar 1987. (Cit  en pages 20 et 21.)
- [Fazenda 2006] N. Fazenda. *Calibration of High-Precision Flexure Parallel Robots*. PhD thesis, Ecole Polytechnique F d rale de Lausanne, 2006. (Cit  en page 20.)
- [Ferraresi 2004] C. Ferraresi, M. Paoloni, S. Pastorelli et F. Pescarmona. *A new 6-DOF parallel robotic structure actuated by wires : the WiRo-6.3*. Journal of Robotics Systems, vol. 21, pages 581–595, 2004. (Cit  en page 15.)
- [Ferson 2002] S. Ferson. Ramas risk calc 4.0 software : Risk assessment with uncertain numbers. Lewis Publishers, 2002. (Cit  en page 27.)

- [Francois 2002] P. Francois. *Parallel mechanisms and redundancy*. In 1st Int. Colloquium, Collaborative Research Centre 562, pages 261–277, 2002. (Non cité.)
- [Gallina 2002] P. Gallina et G. Rosati. *Manipulability of a planar wire driven haptic device*. Mechanism and Machine Theory, vol. 37, pages 215–228, 2002. (Cité en page 17.)
- [Gau 1999] C. Gau et M. A. Stadtherr. Nonlinear parameter estimation using interval analysis. 1999. (Non cité.)
- [Gautier 2001] M. Gautier et Ph. Poignet. *Extended Kalman filtering and weighted least squares dynamic identification of robot*. Control Engineering Practice, vol. 9, no. 12, pages 1361 – 1372, 2001. (Cité en page 27.)
- [Gay 1988] D. M. Gay. *Interval least squares- a diagnostic tool*. pages 183–205. Academic Press Professional, Inc., San Diego, CA, USA, 1988. (Cité en page 43.)
- [Gelman 1995] A. Gelman, J.B. Carlin, H.S. Stern et D.B. Rubin. *Bayesian Data Analysis*. Chapman and Hall. London, 1995. (Cité en page 27.)
- [Goldsztejn 2007] A. Goldsztejn. *Comparison of the Hansen-Sengupta and the Frommer-Lang-Schnurr existence tests*. Computing, vol. 79, pages 53–60, 2007. (Cité en page 107.)
- [Gosselin 2012] C. Gosselin, P. Ren et S. Foucault. *Dynamic trajectory planning of a two-DOF cable-suspended parallel robot*. In ICRA, pages 1476–1481, 2012. (Cité en page 17.)
- [Gouttefarde 2006] M. Gouttefarde, J.-P. Merlet et D. Daney. *Determination of the wrench-closure workspace of 6-DOF parallel cable-driven mechanisms*. In Jadran Lennarcic et B. Roth, éditeurs, Advances in Robot Kinematics, pages 315–322. Springer Netherlands, 2006. (Cité en pages 17 et 28.)
- [Gower 2004] J.C Gower et G.B. Dijksterhuis. Procrustes problems. 2004. (Non cité.)
- [Granvilliers 2012] L. Granvilliers. *Adaptive Bisection of Numerical CSPs*. In Michela Milano, éditeur, Principles and Practice of Constraint Programming, Lecture Notes in Computer Science, pages 290–298. Springer Berlin Heidelberg, 2012. (Cité en page 39.)
- [Guenette 2011] R. Guenette. *Cours sur le préconditionnement, MAT-17992*, 2011. (Cité en page 34.)
- [Hansen 1967] E. Hansen et R. Smith. *Interval arithmetic in matrix computations. II*. Numerical Analysis, vol. 4, pages 1–9, 1967. (Cité en page 44.)
- [Hansen 2003] E. R Hansen. Global optimization using interval analysis. Marcel Dekker Inc., 2003. (Cité en pages 28, 30, 33, 34 et 106.)
- [Higham 2002] N. J. Higham. Accuracy and stability of numerical algorithms. 2002. (Cité en page 44.)
- [Hollerbach 1996] J. M. Hollerbach et C. W. Wampler. *The Calibration Index and Taxonomy for Robot Kinematic Calibration Methods*. The International Journal of Robotics Research, vol. 15, no. 6, pages 573 – 591, dec 1996. (Cité en page 86.)
- [Horst 1996] R. Horst et H. Tuy. Global optimization : Deterministic approaches. Springer, 1996. (Cité en page 38.)

- [Irvine 1974] H. M. Irvine et T. K. Caughey. *The Linear Theory of Free Vibrations of a Suspended Cable*. Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences, vol. 341, no. 1626, pages pp. 299–315, 1974. (Cité en pages 64 et 113.)
- [Iurascu 2003] C. C. Iurascu et F. C. Park. *Geometric Algorithms for Kinematic Calibration of Robots Containing Closed Loops*. Journal of Mechanical Design, vol. 125, no. 1, pages 23–32, 2003. (Cité en page 21.)
- [Jacquez 1985] J. A. Jacquez et P. Greif. *Numerical parameter identifiability and estimability : Integrating identifiability, estimability, and optimal sampling design*. Elsevier, 1985. (Cité en page 85.)
- [Jahn 1974] K. -U Jahn. *Eine Theorie der Gleichungssysteme mit Intervall-Koeffizienten*. ZAMM - Journal of Applied Mathematics and Mechanics / Zeitschrift für Angewandte Mathematik und Mechanik, vol. 54, no. 6, pages 405–412, 1974. (Non cité.)
- [Jaulin 2001] L. Jaulin, M. Kieffer, O. Didrit et E. Walter. Applied interval analysis. Springer, 2001. (Cité en pages 28, 43, 104 et 106.)
- [Jaulin 2009a] L. Jaulin. *Robust set-membership state estimation ; application to underwater robotics*. Automatica, vol. 45, no. 1, pages 202–206, jan 2009. (Cité en page 37.)
- [Jaulin 2009b] Luc Jaulin. *Image shape extraction using interval methods*. In System Identification, volume 15, pages 378–383, 2009. (Cité en page 130.)
- [Jaulin 2010] Luc Jaulin et Gilles Chabert. *Resolution of nonlinear interval problems using symbolic interval arithmetic*. Eng. Appl. Artif. Intell., vol. 23, no. 6, pages 1035–1040, sep 2010. (Cité en page 58.)
- [Joubair 2012] A. Joubair. *Contribution à l'amélioration de la précision absolue des robots parallèles*. PhD thesis, Université du Québec, 2012. (Cité en page 18.)
- [Kawamura 1995] S. Kawamura, W. Choe, S. Tanaka et Pandian. S. *Development of an ultrahigh speed robot FALCON using wire drive system*. In IEEE International Conference on Robotics and Automation, Nagoya, Japan, volume 1, pages 215–220, 1995. (Cité en pages 12 et 15.)
- [Kawamura 2000] S. Kawamura, H. Kino et C. Won. *High-speed manipulation by using parallel wire-driven robots*. Robotica, vol. 18, pages 13–21, 2000. (Cité en page 12.)
- [Kearfott ] R.B. Kearfott. *An interval Newton method for nonlinear Least Squares*. (Non cité.)
- [Kearfott 1990] R.B. Kearfott et M. Novoa III. *INTBIS, a portable interval Newton/Bisection package*. ACM Trans. on Mathematical Software, vol. 16, no. 2, pages 152–157, 1990. (Cité en pages 34 et 39.)
- [Kearfott 1996] R. B Kearfott. Rigorous global search : Continuous problems. Kluwer Academic Publishers, 1996. (Non cité.)
- [Kentel 2005] E. Kentel et M . M . Aral. *2D Monte Carlo versus 2D Fuzzy Monte Carlo health risk assessment*. Stochastic Environmental Research and Risk Assessment, 2005. (Cité en page 28.)
- [Khalil 1991] W. Khalil, M. Gautier et Ch. Enguehard. *Identifiable Parameters and Optimum Configurations for Robots Calibration*. Robotica, vol. 9, no. 01, pages 63–70, 1991. (Cité en pages 84, 85 et 106.)

- [Khalil 1999] W. Khalil et E. Dombre. Modelisation, identification et commande des robots. 1999. (Cit  en pages ix, 6 et 84.)
- [Kieffer ] M. Kieffer et E. Walter. Draft article (For engineering journals) interval analysis for guaranteed nonlinear parameter and state estimation. (Non cit .)
- [Kolev 2001] L. V Kolev. *Automatic Computation of a Linear Interval Enclosure*. Reliable Computing, vol. 7, pages 17–28, 2001. (Non cit .)
- [Kosowski 2002] C. Kosowski et L. Notash. *CAT4 : a novel 4DOF cable actuated parallel manipulator*. 2002. (Cit  en page 15.)
- [Kozak 2006] K. Kozak et Q. Z. E. J. Wang. *Static analysis of cable-driven manipulators with non-negligible cable mass*. IEEE Transactions on Robotics, pages 425–433, 2006. (Cit  en pages 9, 11, 16 et 64.)
- [Kreinovich 1997] V. Kreinovich, A. V Lakeyev, J. Rohn et P. T Kahl. Computational complexity and feasibility of data processing and interval computations. Kluwer, 1997. (Cit  en pages 31 et 43.)
- [Lagrange 2005] S. Lagrange. *Contributions aux m thodes d’estimation en aveugle*, 2005. (Cit  en pages 85 et 106.)
- [Lagrange 2008] S. Lagrange, N. Delanoue et L. Jaulin. *Injectivity analysis using interval analysis : Application to structural identifiability*. Automatica, vol. 44, no. 11, pages 2959–2962, 2008. (Non cit .)
- [Lamaury 2012] J. Lamaury, M. Gouttefarde, M. Michelin et O. Tempier. *Design and Control of a Redundant Suspended Cable-Driven Parallel Robots*. In Latest Advances in Robots Kinematics, pages 237–244, Autriche, 2012. Springer. (Cit  en page 12.)
- [Lamaury 2013] J. Lamaury et M. Gouttefarde. *A Tension Distribution Method with Improved Computational Efficiency*. In Tobias Bruckmann et Andreas Pott,  diteurs, Cable-Driven Parallel Robots, volume 12 of *Mechanisms and Machine Science*, pages 71–85. Springer Berlin Heidelberg, 2013. (Cit  en page 12.)
- [Landsberger 1985] S.E. Landsberger et T.B. Sheridan. *A new design for parallel link manipulators*. In International Conference on Cybernetics and Society, Tucson, USA, pages 81–88, 1985. (Cit  en page 12.)
- [Landsberger 1992] S.E. Landsberger et P. Shanmugasundram. *Workspace of a Parallel Link Crane*. In International Symposium on Robotics, Mechatronics and Manufacturing Systems, pages 479–486, 1992. (Cit  en page 17.)
- [Lau 2011] D. Lau, D. Oetomo et S. K. Halgamuge. *Wrench-Closure Workspace Generation for Cable Driven Parallel Manipulators Using a Hybrid Analytical-Numerical Approach*. Journal of Mechanical Design, vol. 133, no. 7, page 071004, 2011. (Cit  en page 17.)
- [Lee 2004] P.M. Lee. *Bayesian Statistics : An introduction*. Hodder Arnold. London., 2004. (Cit  en page 27.)
- [Legendre 1825] A.-M. Legendre. Traite des fonctions elliptiques et des integrales eul riennes. Huzard-courcier, 1825. (Cit  en page 24.)
- [Lhomme 1993] O. Lhomme. *Consistency Techniques for Numeric CSPs*. In IJCAI, pages 232–238, 1993. (Cit  en page 37.)

- [Lim 2009] W. B Lim, G. Yang, S. H Yeo, S. K Mustafa et I. M Chen. *A generic tension-closure analysis method for fully-constrained cable-driven parallel manipulators*. In Robotics and Automation, 2009. ICRA '09. IEEE International Conference on, pages 2187–2192, may 2009. (Cité en page 16.)
- [Lin 2004] Y. Lin et M. Stadtherr. *LP Strategy for the Interval-Newton Method in Deterministic Global Optimization*. Ind. eng. chemistry research, vol. 43, pages 3741–3749, 2004. (Non cité.)
- [Little 2010] M.P. Little, W.F. Heidenreich et G. Li. *Parameter identifiability and redundancy : theoretical considerations*. PloS one, 2010. (Cité en pages 85 et 86.)
- [Lubrano 2011] E. Lubrano, M. Bouri et R. Clavel. *Ultra-high-precision Industrial Robots Calibration*. In IEEE International Conference on Robotics and Automation, pages 228 – 233, Shanghai, may 2011. (Cité en page 21.)
- [Manteuffel 1981] T. A. Manteuffel. *An Interval Analysis Approach to Rank Determination in Linear Least Squares Problems*. SIAM Journal on Scientific and Statistical Computing, vol. 2, no. 3, pages 335–348, 1981. (Cité en page 43.)
- [Markov 1886] A. Markov. *Sur une question de maximum et de minimum proposée par M. Tchebycheff*. Acta Mathematica, 1886. (Cité en page 28.)
- [Marquet 2002] F. Marquet. *Contribution à l'étude de l'apport de la redondance en robotique parallèle*. PhD thesis, 2002. (Cité en page 21.)
- [Maurine 1996] P. Maurine et E. Dombre. *A calibration procedure for the parallel robot Delta 4*. In ICRA, pages 975–980, 1996. (Cité en page 22.)
- [McColl 2011] D. McColl et L. Notash. *Workspace formulation of planar wire-actuated parallel manipulators*. Robotica, vol. 29, no. 04, pages 607–617, 2011. (Cité en page 17.)
- [Meggiolaro 2000] M.A. Meggiolaro et S. Dubowsky. *An analytical method to eliminate the redundant parameters in robot calibration*. IEEE International Conference on Robotics and Automation, 2000. (Cité en page 84.)
- [Meng 2003] G. Meng, L. Tiemin et Y. Wensheng. *Calibration method and experiment of Stewart platform using a laser tracker*. In Systems, Man and Cybernetics, 2003. IEEE International Conference on, volume 3, pages 2797 – 2802, oct 2003. (Cité en page 21.)
- [Merlet 1997] J.P. Merlet. Les robots parallèles. Hermes Science Publishing, 1997. (Cité en pages 6, 8 et 65.)
- [Merlet 2004a] J.-P. Merlet. *Analysis of the Influence of Wires Interference on the Workspace of Wire Robots*. Advances in Robot Kinematics, pages 211–218, June 2004. (Cité en page 17.)
- [Merlet 2004b] J. P. Merlet. *Solving the Forward Kinematics of a Gough-Type Parallel Manipulator with Interval Analysis*. The international Journal of Robotics Research, vol. 23, no. 3, pages 221–235, 2004. (Cité en page 28.)
- [Merlet 2006] J.P. Merlet. Parallel robots, 2nd edition, volume 128. Springer, 2006. (Cité en pages 6, 23 et 83.)
- [Merlet 2008a] J-P. Merlet et D. Daney. *Appropriate Design of Parallel Manipulators*. Smart Devices and machines for advanced manufacturing, pages 1–25, 2008. (Non cité.)

- [Merlet 2008b] J.P. Merlet. *Kinematics of the wire-driven parallel robot MARIO-NET using linear actuators*. In IEEE Int. Conf. on Robotics and Automation, Pasadena, 2008. (Cit  en pages 9, 12, 15 et 63.)
- [Merlet 2009] J.-P. Merlet. *Analysis of wire elasticity for wire driven parallel robots*. 2009. (Cit  en page 16.)
- [Merlet 2012a] J.-P. Merlet. *The kinematics of the redundant  $N - 1$  wire driven parallel robot*. In ICRA, pages 2313–2318, 2012. (Cit  en page 66.)
- [Merlet 2012b] J.-P. Merlet. *Wire-Driven parallel robot : open issues*. In RoManSy, Paris (France), 2012. (Cit  en pages 17, 18 et 68.)
- [Miermeister 2012] P. Miermeister et A. Pott. *Auto Calibration Method for Cable-Driven Parallel Robots Using Force Sensors*. In J. Lenarcic et M. Husty,  diteurs, Latest Advances in Robot Kinematics, pages 269–276. Springer Netherlands, 2012. (Cit  en pages 22 et 68.)
- [Ming 1994] A Ming et T Higuchi. *Study On Multiple Degree-Of-Freedom Positioning Mechanism Using Wires .2. Development Of A Planar Completely Restrained Positioning Mechanism*. International Journal Of The Japan Society For Precision Engineering, vol. 28, no. 3, pages 235–242, Sep 1994. (Cit  en page 12.)
- [Montgomery 2009] V. Montgomery. *New statistical methods in risk assessment by probability bounds*. PhD thesis, Durham University, 2009. (Cit  en page 27.)
- [Moore 1966] R. E Moore. Interval analysis. Prentice-Hall, 1966. (Cit  en pages 28, 29, 38 et 106.)
- [Nahvi 1996] A. Nahvi et J. M. Hollerbach. *The Noise Amplification Index for Optimal Pose Selection in Robot Calibration*. 1996. (Cit  en pages 84, 86 et 87.)
- [Neumaier 1990] A. Neumaier. Interval methods for systems of equations. Cambridge Univ. Press, 1990. (Cit  en pages 28, 30, 33, 43, 106 et 107.)
- [Neumaier 2000] A. Neumaier. *A simple derivation of the Hansen-Blick-Rohn-Ning-Kearfott enclosure for linear interval equations*, 2000. (Cit  en page 34.)
- [Oetomo 2009] D. Oetomo, D. Daney, B. Shirinzadeh et J.-P. Merlet. *An Interval-Based Method for Workspace Analysis of Planar Flexure-Jointed Mechanism*. Journal of Mechanical Design, vol. 131, no. 1, page 011014, 2009. (Non cit .)
- [Oettli 1964] W. Oettli et W Prager. *Computability of approximate solution of linear equations with given bounds for coefficients right-hand sides*. Numerische Mathematik, no. 6, pages 405–9, 1964. (Cit  en page 33.)
- [Oettli 1965] W. Oettli. *On the Solution Set of a Linear System with Inaccurate Coefficients*. Journal of the Society for Industrial and Applied Mathematics Series B Numerical Analysis, vol. 2, no. 1, pages 115–118, 1965. (Cit  en pages 33 et 35.)
- [Ogahara 2003] Y. Ogahara, Y. Kawato, K. Takemura et T. Naeno. *A wire-driven miniature five fingered robot hand using elastic elements as joints*. In International Conference on Intelligent Robots and Systems, Las Vegas, USA, pages 2672–2677, 2003. (Cit  en page 15.)
- [Oh 2005a] s. Oh, K. K. Makala et Albus J. S. Agrawal S. *A dual-stage planar cable robot : Dynamic modeling and design of a robust controller with positive*

- inputs*. ASME Journal of Mechanical Design, vol. 127, pages 612–620, 2005. (Cité en page 15.)
- [Oh 2005b] s. Oh, K. K. Makala et S. Agrawal. *Dynamic modeling and robust controller design of a two-stage parallel cable robot*. Multibody System Dynamics, vol. 13, pages 385–399, 2005. (Cité en page 17.)
- [Ollivier 2000] F. Ollivier. *Identifiabilité et identification : du Calcul Formel au Calcul Numérique*. ESAIM, 2000. (Cité en pages 85, 86 et 88.)
- [Ottaviano 2006] E. Ottaviano et M. Ceccarelli. *Numerical and experimental characterization of singularities of a six-wire parallel architecture*. Robotica, vol. 25, no. 3, pages 315–324, may 2006. (Cité en page 17.)
- [Park 1995] F. C. Park. *Distance metrics on the rigid-body motions with applications to mechanism design*. American Society of Mechanical Engineers, 1995. (Non cité.)
- [Patel 2000] A. J. Patel et K. F. Ehmann. *Calibration of a hexapod machine tool using a redundant leg*. International Journal of Machine Tools and Manufacture, vol. 40, no. 4, pages 489–512, mar 2000. (Cité en pages 21, 22, 26 et 82.)
- [Popova 2005] E. D. Popova. *Improved Solution Enclosures for Over- and Under-determined Interval Linear Systems*. In Lecture Notes in Computer Science, Proc. 5th Int. Conf. LSSC, Sozopol, 2005. (Cité en pages 43 et 44.)
- [Porta 2005a] J.M. Porta, L. Ros, F. Thomas et C. Torras. *A branch-and-prune solver for distance constraints*. Robotics, IEEE Transactions on, vol. 21, no. 2, pages 176 – 187, avril 2005. (Non cité.)
- [Porta 2005b] J.M. Porta, L. Ros, F. Thomas et C. Torras. *A branch-and-prune solver for distance constraints*. Robotics, IEEE Transactions on, vol. 21, no. 2, pages 176 – 187, avril 2005. (Non cité.)
- [Pott 2006] A. Pott et M. Hiller. *A Parallel Implementation for the Optimization of Parallel Kinematic Machines under Process Requirements*. In EuCoMeS, Obergurgl (Austria), 2006. (Non cité.)
- [Pusey 2004] j. Pusey, A. Fattah, S. Agrawal et E. Messina. *Design and workspace analysis of a 6-6 cable-suspended parallel robot*. Mechanisms and Machine Theory, vol. 39, pages 761–778, 2004. (Cité en page 15.)
- [Renaud 2002] P. Renaud, N. Andreff, P. Martinet, G. Gogu et M. Dhome. *Identification géométrique de mécanismes parallèles par vision*. 2002. (Cité en page 84.)
- [Renaud 2003] P. Renaud. *Apport de la vision pour l'identification géométrique de mécanismes parallèles*. PhD thesis, Université Blaise Pascal, Clermont-Ferrand, 2003. (Cité en page 20.)
- [Renaud 2005] P. Renaud, N. Andreff, P. Martinet et G. Gogu. *Kinematic Calibration of Parallel Mechanisms : A Novel Approach Using Legs Observation*. Robotics, IEEE Transactions on, vol. 21, no. 4, pages 529 – 538, aug. 2005. (Cité en page 21.)
- [Riehl 2009] N. Riehl, M. Gouttefarde, S. Krut, C. Baradat et F. Pierrot. *Effects of non-negligible cable mass on the static behavior of large workspace cable-driven parallel mechanisms*. In Robotics and Automation, 2009. ICRA '09.

- IEEE International Conference on, pages 2193–2198, may 2009. (Cité en pages 16 et 69.)
- [Riehl 2011] N. Riehl. *Modélisation et conception de robots parallèles à câbles*. PhD thesis, Université Montpellier II, 2011. (Cité en pages 12, 80 et 113.)
- [Roberts 1998] R. G. Roberts, T. Graham et T. Lippitt. *On the Inverse Kinematics, Statics, and Fault Tolerance of Cable-Suspended Robots*. Journal of Robotic Systems, vol. 15, page 581597, 1998. (Cité en pages 16 et 66.)
- [Rohn 1985] J. Rohn. *Inner solutions of linear interval systems*. In Proceedings of the International Symposium on interval mathematics on Interval mathematics 1985, pages 157–158, London, UK, UK, 1985. Springer-Verlag. (Cité en page 35.)
- [Rohn 1994] J. Rohn. *NP-hardness Results for Linear Algebraic Problems with Interval Data*. pages 463–471. Elsevier, 1994. (Cité en pages 31 et 33.)
- [Rohn 2012] J. Rohn. *A Handbook of Results on Interval Linear Problems*. Rapport technique, Academy of Sciences of the Czech Republic, Prague, 2012. (Cité en pages 107 et 110.)
- [Rump 1987] S.M. Rump. *Solution of Linear Systems with Verified Accuracy*. Applied Numerical Mathematics, 1987. (Cité en pages 30, 33 et 44.)
- [Sandretto 2012a] J. Alexandre Dit Sandretto, D. Daney et M. Gouttefarde. *Calibration of a Fully-Constrained Parallel Cable-Driven Robot*. In RoManSy, Paris (France), 2012. (Cité en pages 82 et 97.)
- [Sandretto 2012b] J. Alexandre Dit Sandretto, G. Trombettoni et D. Daney. *Confirmation of Hypothesis on Cable Properties for Cable-driven Robots*. In Eucomes, Santander (Spain), 2012. (Non cité.)
- [Sato 2002] M. Sato. *Development of string-based force display : SPIDAR*. In 8th International Conference on Virtual Systems and Multimedia. Citeseer, 2002. (Cité en page 15.)
- [Schön 2005] S. Schön et H. Kutterer. *Using Zonotopes for Overestimation-Free Interval Least-Squares-Some Geodetic Applications*. Reliable Computing, vol. 11, pages 137–155, 2005. 10.1007/s11155-005-3034-4. (Cité en page 43.)
- [Shary 2002] S.P. Shary. *A New Technique in Systems Analysis Under Interval Uncertainty and Ambiguity*. Reliable Computing, vol. 8, pages 321–418, 2002. (Cité en pages 31, 34 et 48.)
- [Siciliano 2008] B. Siciliano et O. Khatib. Springer handbook of robotics. Springer, 2008. (Cité en page 18.)
- [sky 2007] *Skycam*, 2007. (Cité en page 11.)
- [Stump 2006] E. Stump et V. Kumar. *Workspace of cable-actuated parallel manipulators*. ASME Journal of Mechanical Design, vol. 128, pages 159–167, 2006. (Cité en page 17.)
- [Su 2011] K.-H. Su. *Fuzzy Model Identification With Enhanced Validity Criterion for Mechanical System Design*. Journal of Mechanical Design, vol. 133, no. 10, page 104501, 2011. (Cité en page 28.)
- [Sun 2008] Y. Sun et J.M. Hollerbach. *Observability index selection for robot calibration*. 2008. (Cité en pages 84, 86 et 87.)



- [Tadokoro 1999] S. Tadokoro, R. Verhoeven, M. Hiller et T. Takamori. *A portable parallel manipulator for search and rescue at large-scale urban earthquakes and an identification algorithm for the installation in unstructured environments*. In Intelligent Robots and Systems, volume 2, pages 1222–1227 vol.2, 1999. (Cit  en page 22.)
- [Takeda 2004] Y. Takeda, G. Shen et H. Funabashi. *A DBB-Based Kinematic Calibration Method for In-Parallel Actuated Mechanisms Using a Fourier Series*. Journal of Mechanical Design, vol. 126, no. 5, pages 856–865, 2004. (Cit  en pages 21, 22 et 86.)
- [Trombettoni 2007] G. Trombettoni et G. Chabert. *Constructive Interval Disjunction*. In Proc. CP, LNCS 4741, pages 635–650, 2007. (Cit  en page 37.)
- [Trombettoni 2011] G. Trombettoni, I. Araya, B. Neveu et G. Chabert. *Inner Regions and Interval Linearization for Global Optimization*. In AAAI 2011, San Francisco, CA, USA, 2011. (Cit  en page 39.)
- [Tsai 1996] L-W. Tsai. *Kinematics of A Three DOF Platform With Three Extensible Limbs*. In Recent Advances in Robot Kinematics, pages 401–410, 1996. (Cit  en page 16.)
- [Ulam 1949] S. Ulam. *The Monte Carlo Method*. Journal of the American Statistical Association, 1949. (Cit  en page 28.)
- [Varziri 2007] M. S. Varziri et L. Notash. *Kinematic calibration of a wire-actuated parallel robot*. Mechanism and Machine Theory, vol. 42, no. 8, pages 960 – 976, 2007. (Cit  en pages 22 et 62.)
- [Verhoeven 1998] R. Verhoeven, M. Hiller et S. Tadokoro. *Workspace, stiffness, singularities and classification of tendon-driven stewart platforms*. In Advances in Robot Kinematics Analysis and Control, pages 105–114, 1998. (Cit  en page 17.)
- [Verhoeven 2000] R. Verhoeven et M. Hiller. *Estimating the controllable workspace of tendon-based Stewart platforms*. In Advances in Robot Kinematics, pages 277–284, 2000. (Cit  en page 17.)
- [Vischer 1996] P. Vischer. *Improving the Accuracy of Parallel Robots*. PhD thesis, Ecole Polytechnique F d rale de Lausanne, 1996. (Cit  en page 20.)
- [Vischer 1998] P. Vischer et R. Clavel. *Kinematic calibration of the parallel Delta robot*. Robotica, vol. 16, pages 207–218, 1998. (Cit  en page 21.)
- [Walter 1997] E. Walter et L. Pronzat. *Identification of parametric models from experimental data*. Springer, 1997. (Cit  en page 85.)
- [Walter 2004] E. Walter, I. Braems, L. Jaulin et M. Kieffer. *Guaranteed Numerical Computation as an Alternative to Computer Algebra for Testing Models for Identifiability*. In Ren  Alt, Andreas Frommer, R.Baker Kearfott et Wolfram Luther,  diteurs, Numerical Software with Result Verification, volume 2991 of *Lecture Notes in Computer Science*, pages 124–131. Springer Berlin Heidelberg, 2004. (Non cit .)
- [Walter 2007]  . Walter et M. Kieffer. *Guaranteed nonlinear parameter estimation in knowledge-based models*. Journal of Computational and Applied Mathematics, vol. 199, no. 2, pages 277 – 285, 2007. (Cit  en page 104.)

- [Wampler 1992] C. Wampler et T. Arai. *Calibration of robots having kinematic closed loops using nonlinear least-squares estimation*. Proc. IFToMM-jc Int. Symp. Theory Mach. Mech., pages 153 – 158, 1992. (Cit  en pages 23 et 24.)
- [Wampler 1995] C. W. Wampler, J. M. Hollerbach et T. Arai. *An Implicit Loop Method for Kinematic Calibration and Its Application to Closed-Chain Mechanisms*. IEEE Transactions on Robotics and Automation, vol. 11, no. 5, pages 710–724, oct 1995. (Cit  en pages 20, 21 et 23.)
- [Wang 1993] J. Wang et O. Masory. *On the accuracy of a Stewart platform - Part I. The effect of manufacturing tolerances*. In Robotics and Automation, volume 1, pages 114 –120, may 1993. (Cit  en pages 19 et 20.)
- [Williams 2003] R.L. Williams, J.S. Albus et R. Bostelman. *3D cable-based cartesian metrology system*. 2003. (Cit  en page 15.)
- [Zhuang 1994] H. Zhuang, K. Wang et Z.S. Roth. *Optimal selection of measurement configurations for robot calibration using simulated annealing*. In Robotics and Automation, 1994. Proceedings., 1994 IEEE International Conference on, pages 393 –398 vol.1, may 1994. (Cit  en pages 28 et 87.)
- [Zhuang 1995] H. Zhuang, O. Masory et Jiahua Y. *Kinematic calibration of a Stewart platform using pose measurements obtained by a single theodolite*. In Intelligent Robots and Systems 95. 'Human Robot Interaction and Cooperative Robots', Proceedings. 1995 IEEE/RSJ International Conference on, volume 2, pages 329 –334 vol.2, aug 1995. (Cit  en page 21.)
- [Zhuang 1997] H. Zhuang. *Self-Calibration of Parallel Mechanisms with a Case Study on Stewart Platforms*. Robotics and Automation, IEEE Transactions on, vol. 13, no. 3, pages 387 –397, jun 1997. (Non cit .)
- [Zhuang 1998] H. Zhuang, J. Yan et O. Masory. *Calibration of stewart platforms and other parallel manipulators by minimizing inverse kinematic residuals*. Journal of Robotic Systems, vol. 15, no. 7, pages 395–405, 1998. (Cit  en page 21.)
- [Zitzewitz 2009] J. Zitzewitz, G. Rauter et A. Brunschweiler. *A versatile wire robot concept as a haptic interface for sport simulation*. 2009. (Cit  en page 15.)



# Procédure d'étalonnage

---

## A.1 Introduction

Cette annexe a pour vocation à décrire et encadrer toute la procédure d'acquisition des mesures pour l'étalonnage du robot à câbles Cogiro. Il ne développe pas le processus d'identification des paramètres qui se fera en mode hors ligne ultérieurement.

## A.2 Préparation

Le robot considéré est complexe par son architecture et son instrumentation. En effet, la commande peut être couplée à un réseau de caméras important, les capteurs sur le robots sont nombreux (capteurs de tension, codeurs moteur). L'utilisation de ce robot fait donc intervenir toutes les équipes qui participent au projet. À cela s'ajoute également une sécurité drastique due à la taille du robot ainsi qu'aux tensions qu'il est capable de générer dans les câbles. Une bonne préparation de l'acquisition des mesures d'étalonnage est donc essentielle afin de réduire le temps d'immobilisation du robot (destiné à l'industrie) tout en garantissant la qualité des données recueillies.

Le processus de préparation se déroulera comme suit :

1. Consigner tous les paramètres géométriques et statiques de la structure du robot et des capteurs ;
2. Monter les cibles de manière fixe et répétable (sur la plate-forme pour le laser et les caméras, au sol ou sur la base pour les positions de référence) ;
3. Étalonner les systèmes d'acquisition externes (laser, caméras) ;
4. Définir les poses d'étalonnage du robot (ceci peut avoir été prévu en amont si les données du robot sont connues et suffisamment proche de la réalité) ;
5. Définir les poses de vérification (les poses qui serviront à valider l'amélioration théorique de la précision du robot) ;
6. Vérifier l'atteignabilité de toutes les poses définies ;
7. Pratiquer l'acquisition des mesures sur une pose ;
8. Vérifier l'acquisition des données et leur format ;
9. Vérifier la répétabilité des mesures sur la même pose.

Ces différentes phases, si elles sont effectuées dans l'ordre peuvent améliorer fortement la qualité des mesures obtenues.

## A.3 Acquisition des mesures

### A.3.1 Diagnostic comportemental

Après avoir préparé l'acquisition des mesures via les différentes phases décrites précédemment, une dernière vérification de la concordance du comportement et du modèle peut être effectuée. Celle-ci consiste simplement à vérifier sur quelques poses la bonne approximation du modèle (ou de la commande) dont on veut identifier les paramètres. Ceci peut être fait par la vérification de la concordance entre les valeurs de la commande (longueurs des câbles, pose demandée) et des mesures externes (mesure de la pose par le laser par exemple). Cette étape peut permettre d'éviter de faire des mesures inutiles si par exemple le modèle d'enroulement ne correspond pas suffisamment à la réalité, dans ce cas les mesures de longueur des câbles seraient toutes erronées et donc sans intérêt. La cohérence de la numérotation (câbles, caméras, etc...) et de l'orientation des divers repères se doit également d'être vérifiée.

### A.3.2 Acquisition

Les mesures sont de deux types, il y a celles de l'environnement et celles des poses du robot.

#### A.3.2.1 Environnement

Les mesures de l'environnement permettent d'initialiser les paramètres extérieurs au robot (température ambiante, luminosité, etc...) et de définir la position des différents instruments de mesure. Elles doivent être répétées tout au long de la prise de mesure d'étalonnage et à chaque déplacement des appareils de mesures (modification de la cible ou déplacement du laser pour améliorer la visibilité par exemples) ou de modification de l'environnement (ouverture d'une fenêtre par exemple).

#### A.3.2.2 Poses

Les poses, même si elles sont destinées à deux usages différents, identification des paramètres et vérification, sont du même type et doivent être prises de la même façon. Idéalement, elles doivent être réalisées alternativement ou de manière aléatoire afin de ne pas subir une éventuelle divergence des mesures ou de la commande. Elles se font uniquement lorsque le robot est correctement positionné et lorsque l'équilibre statique est atteint.

## A.4 Précautions particulières

À chaque déplacement des instruments de mesures, il est impératif de reprendre les mesures nécessaires à leur localisation.

Obtenir pour chaque mesure, instrument ou paramètre une borne sur sa tolérance en vue de l'utilisation de l'analyse par intervalles.

Il peut être très utile de conserver un ordre voire une chronologie dans l'acquisition des mesures.

# Caractéristiques du prototype ReelAx8

---

Les paramètres géométriques théoriques définissant le bâti sont rassemblés dans la table B.1.

	$a_1$	$a_2$	$a_3$	$a_4$	$a_5$	$a_6$	$a_7$	$a_8$
x	1.155	1.208	-1.129	-1.152	-1.104	-1.209	1.193	1.165
y	-1.869	-1.823	-1.796	-1.854	1.780	1.799	1.745	1.833
z	2.718	.357	2.701	.415	2.735	.428	2.717	.393

TAB. B.1 – Les paramètres géométriques théoriques du bâti du robot ReelAx8

Les paramètres géométriques théoriques définissant la plateforme sont rassemblés dans la table B.2.

	$b_1$	$b_2$	$b_3$	$b_4$	$b_5$	$b_6$	$b_7$	$b_8$
x	-0.239	-0.234	0.178	0.185	0.189	0.190	-0.232	-0.230
y	-0.180	-0.184	-0.184	-0.188	0.187	0.180	0.191	0.188
z	0.186	-0.162	0.238	-0.233	0.235	-0.236	0.184	-0.164

TAB. B.2 – Les paramètres géométriques théoriques de la plateforme du robot ReelAx8

Les câbles sont en aciers. Leur masse linéique est donnée par le constructeur à  $m = 0.007kg/m$  et la raideur  $k = 137000N/m$ . En utilisation normale, les tensions dans les câbles sont en théorie comprises entre  $20N$  et  $120N$ . L'espace de travail en position est un parallélépipède centré de  $[1, 3] \times [1, 2] \times [1, 2]$ . En rotation, il est possible en théorie de couvrir de  $-5$  à  $+5$  degrés sur chaque orientations.



# Caractéristiques du robot CoGiRo

---

Les paramètres géométriques théoriques définissant le bâti sont rassemblés dans la table C.1.

	$a_1$	$a_2$	$a_3$	$a_4$	$a_5$	$a_6$	$a_7$	$a_8$
x	7.15	7.405	- 7.15	- 7.405	7.15	7.405	- 7.15	- 7.405
y	5.405	5.15	5.405	5.15	- 5.405	- 5.15	- 5.405	- 5.15
z	5.531	5.521	5.531	5.521	5.531	5.521	5.531	5.521

TAB. C.1 – Les paramètres géométriques théoriques du bâti du robot CoGiRo

Les paramètres géométriques théoriques définissant la plateforme sont rassemblés dans la table C.2.

	$b_1$	$b_2$	$b_3$	$b_4$	$b_5$	$b_6$	$b_7$	$b_8$
x	- 0.266	0.341	0.356	- 0.5	- 0.356	0.5	0.266	- 0.341
y	0.5	- 0.5	0.492	- 0.5	- 0.492	0.5	- 0.5	0.5
z	1.	2.	2.	1.	2.	1.	1.	2.

TAB. C.2 – Les paramètres géométriques théoriques de la plateforme du robot CoGiRo

Les câbles sont en aciers. Leur masse linéique est donnée par le constructeur à  $m = 0.064kg/m$  et la raideur  $k = 311795N/m$ . En utilisation normale, les tensions dans les câbles sont en théorie comprises entre  $40N$  et  $5000N$  (correspond au dimensionnement maximal du robot, soit  $500kg$  sur un câble). L'espace de travail en position est un parallélépipède centré de  $14m \times 10m$  au sol sur  $5m$  de haut. En rotation, il est possible en théorie de couvrir de  $-0.4$  à  $+0.4$  radians sur chaque orientations (dans certaines poses il est possible d'aller plus loin autour de  $z$ ).





# Listes des poses pour l'étalonnage des robots

---

## D.1 Poses pour le robot ReelAx8

$x$	$y$	$z$	$\psi$	$\theta$	$\phi$	$l_1$	$l_2$	$l_3$	$l_4$	$l_5$	$l_6$	$l_7$	$l_8$
-0.174	0.100	-1.601	-0.535	-1.222	-0.507	2.376	2.380	2.299	2.335	2.273	2.318	2.328	2.358
-0.171	0.103	-1.611	-0.485	-1.207	-0.533	2.373	2.382	2.292	2.344	2.280	2.310	2.331	2.356
-0.180	0.096	-1.587	-0.610	-1.236	-0.464	2.388	2.379	2.312	2.326	2.264	2.325	2.324	2.365
-0.166	0.095	-1.618	-0.512	-1.179	-0.455	2.359	2.390	2.313	2.321	2.291	2.307	2.312	2.372
-0.177	0.104	-1.579	-0.563	-1.281	-0.590	2.397	2.358	2.277	2.358	2.252	2.341	2.349	2.338
-0.172	0.109	-1.601	-0.548	-1.196	-0.521	2.366	2.370	2.307	2.343	2.264	2.309	2.338	2.367
-0.176	0.086	-1.602	-0.513	-1.261	-0.484	2.391	2.394	2.287	2.323	2.286	2.330	2.314	2.345
-0.525	0.453	-1.618	-0.516	-1.167	-0.460	2.681	2.722	2.064	2.073	2.043	2.040	2.646	2.697
-0.535	0.462	-1.579	-0.569	-1.269	-0.592	2.718	2.687	2.032	2.103	2.010	2.072	2.685	2.662
0.188	-0.261	-1.616	-0.514	-1.186	-0.453	2.116	2.133	2.638	2.642	2.609	2.637	2.048	2.117
0.180	-0.253	-1.577	-0.561	-1.289	-0.588	2.147	2.100	2.598	2.682	2.569	2.677	2.082	2.086
-0.519	-0.255	-1.612	-0.489	-1.195	-0.537	2.749	2.753	2.668	2.726	1.964	2.002	2.031	2.044
-0.526	-0.262	-1.592	-0.597	-1.223	-0.477	2.766	2.745	2.690	2.707	1.960	2.011	2.029	2.047
0.178	0.464	-1.610	-0.482	-1.214	-0.530	2.051	2.068	1.974	2.012	2.653	2.678	2.694	2.724
-0.172	0.110	-1.900	-0.556	-1.198	-0.521	2.254	2.514	2.203	2.471	2.153	2.439	2.226	2.505
-0.174	0.087	-1.901	-0.519	-1.267	-0.484	2.276	2.536	2.180	2.450	2.175	2.461	2.202	2.483
-0.171	0.109	-1.304	-0.540	-1.191	-0.521	2.510	2.257	2.444	2.249	2.409	2.213	2.479	2.262
-0.174	0.086	-1.303	-0.505	-1.255	-0.486	2.534	2.280	2.425	2.227	2.429	2.235	2.457	2.238
-0.197	-0.574	-1.871	-0.539	-1.204	-0.512	2.464	2.664	2.859	3.059	2.188	2.453	1.615	1.938
0.551	0.066	-1.916	-0.555	-1.263	-0.460	1.586	1.951	2.253	2.525	2.850	3.110	2.340	2.647
-0.161	0.792	-1.881	-0.551	-1.223	-0.507	2.286	2.549	1.567	1.862	2.343	2.585	2.878	3.092
-0.850	0.095	-1.887	-0.536	-1.208	-0.520	2.924	3.119	2.362	2.610	1.549	1.877	2.292	2.527
-0.178	-0.608	-1.615	-0.539	-1.214	-0.522	2.552	2.552	2.948	2.997	2.300	2.370	1.717	1.759
0.535	0.100	-1.598	-0.535	-1.239	-0.494	1.758	1.760	2.344	2.373	2.923	2.982	2.460	2.511
-0.177	0.817	-1.601	-0.538	-1.218	-0.504	2.414	2.437	1.682	1.708	2.439	2.464	2.983	3.016
-0.878	0.097	-1.604	-0.537	-1.198	-0.517	3.033	3.036	2.465	2.504	1.671	1.690	2.405	2.410
-0.174	0.097	-1.603	-0.528	-1.219	-0.507	2.377	2.380	2.299	2.337	2.272	2.314	2.326	2.356
-0.173	0.097	-1.600	-0.540	-1.222	-0.501	2.377	2.379	2.300	2.334	2.271	2.318	2.326	2.358
-0.172	0.097	-1.603	-0.531	-1.215	-0.499	2.374	2.380	2.300	2.330	2.274	2.315	2.325	2.360
-0.173	0.099	-1.600	-0.536	-1.225	-0.510	2.377	2.377	2.296	2.335	2.271	2.319	2.329	2.357
-0.173	0.100	-1.601	-0.536	-1.215	-0.508	2.374	2.377	2.301	2.337	2.270	2.315	2.329	2.360

-0.173	0.095	-1.601	-0.530	-1.226	-0.501	2.378	2.381	2.297	2.333	2.274	2.319	2.324	2.355
-0.224	-0.226	-1.588	-0.580	-1.208	-0.581	2.477	2.450	2.617	2.684	2.189	2.274	2.056	2.055
0.260	0.173	-1.662	-0.506	-1.257	-0.488	1.959	2.028	2.192	2.274	2.665	2.754	2.423	2.513
-0.180	0.429	-1.547	-0.479	-1.256	-0.499	2.412	2.375	2.007	1.994	2.365	2.346	2.648	2.629
-0.481	-0.009	-1.619	-0.561	-1.234	-0.459	2.670	2.687	2.428	2.457	1.995	2.051	2.229	2.276
-0.208	-0.348	-1.575	-0.598	-1.168	-0.522	2.496	2.460	2.758	2.776	2.230	2.277	1.952	1.960
0.196	0.047	-1.549	-0.558	-1.267	-0.505	2.081	2.028	2.346	2.358	2.617	2.663	2.348	2.350
-0.314	-0.015	-1.646	-0.539	-1.145	-0.475	2.490	2.537	2.440	2.472	2.122	2.167	2.204	2.282
-0.224	-0.253	-1.320	-0.486	-1.239	-0.518	2.624	2.383	2.719	2.576	2.368	2.186	2.176	1.920
0.260	0.159	-1.430	-0.520	-1.221	-0.417	2.077	1.925	2.324	2.171	2.766	2.674	2.504	2.433
-0.177	0.450	-1.361	-0.523	-1.182	-0.513	2.478	2.286	2.133	1.941	2.435	2.264	2.747	2.599

TAB. D.1 – Poses d'étalonnage et de validation du robot ReelAx8 (en mètres et en radians)

## D.2 Poses pour le robot CoGiRo

$x$	$y$	$z$	$\psi$	$\theta$	$\phi$	$l_1$	$l_2$	$l_3$	$l_4$	$l_5$	$l_6$	$l_7$	$l_8$
-3.653	-2.075	1.598	0.004	-0.011	0.399	6.340	5.220	9.243	8.416	14.065	13.078	11.790	11.808
-2.750	-1.186	1.587	-0.006	-0.005	-0.191	7.046	6.648	8.603	8.517	12.514	12.161	11.044	11.502
-3.653	-0.580	1.594	0.005	-0.012	0.478	7.194	6.200	8.064	7.192	13.408	12.263	12.358	12.193
-3.643	-0.872	1.880	-0.365	-0.014	-0.014	6.518	6.484	7.863	7.374	13.246	12.347	11.935	12.416
0.237	-0.881	1.581	-0.001	-0.006	0.397	9.814	8.677	10.444	10.034	10.388	9.429	9.025	8.701
-3.654	-1.484	2.190	0.002	0.412	0.001	6.177	5.779	8.252	8.250	13.275	12.568	11.705	11.562
1.839	-2.190	1.234	-0.011	0.003	-0.999	10.066	10.347	11.972	12.762	9.552	10.220	7.018	7.439
3.652	-2.681	1.982	-0.008	0.002	-0.040	11.987	11.125	13.690	13.937	9.171	9.177	5.370	5.207
3.644	2.190	1.228	0.014	0.330	0.000	13.959	13.649	11.711	12.456	6.251	5.598	9.310	8.557
3.647	-2.181	1.485	-0.003	0.000	0.301	12.410	11.233	13.705	13.565	9.192	8.697	6.101	5.522
-1.660	-1.669	2.475	0.002	0.388	0.004	7.415	6.994	9.264	9.501	11.666	11.094	9.774	9.614
-3.151	2.676	2.470	-0.004	-0.002	-0.501	8.872	9.409	5.278	5.648	11.105	10.836	12.931	13.596

TAB. D.2 – Jeu de poses d'étalonnage du robot CoGiRo  $S_1$  (en mètres et en radians)

$x$	$y$	$z$	$\psi$	$\theta$	$\phi$	$l_1$	$l_2$	$l_3$	$l_4$	$l_5$	$l_6$	$l_7$	$l_8$
-0.000	0.003	1.582	-0.001	-0.000	-0.004	9.789	9.188	9.538	9.595	9.843	9.246	9.486	9.537
-0.000	-0.001	1.583	0.373	0.001	-0.003	9.879	8.923	9.578	9.806	9.802	9.536	9.424	9.410
0.002	0.003	1.570	-0.378	-0.006	-0.009	9.748	9.481	9.483	9.477	9.942	8.989	9.521	9.753
-0.005	-0.003	1.583	-0.001	0.389	0.001	9.848	9.478	9.506	9.966	9.741	9.074	9.586	9.149
0.006	0.005	1.579	-0.002	-0.395	-0.010	9.675	9.021	9.639	9.213	9.898	9.545	9.446	9.918
-0.002	0.006	1.581	0.002	-0.006	0.391	10.026	8.978	9.736	9.364	10.084	9.050	9.685	9.325
0.003	-0.003	1.579	-0.006	-0.000	-0.398	9.533	9.420	9.379	9.790	9.592	9.472	9.317	9.716

TAB. D.3 – Jeu de poses d'étalonnage du robot CoGiRo  $S_2$  (en mètres et en radians)

$x$	$y$	$z$	$\psi$	$\theta$	$\phi$	$l_1$	$l_2$	$l_3$	$l_4$	$l_5$	$l_6$	$l_7$	$l_8$
-5.803	0.083	1.603	-0.001	-0.015	0.193	6.512	6.207	6.778	5.883	14.927	13.983	14.369	14.561
-5.798	0.074	1.616	-0.020	-0.049	-0.395	6.225	6.477	6.525	6.202	14.548	14.291	14.141	14.808
-5.802	0.081	1.599	-0.001	-0.011	-0.006	6.406	6.307	6.696	5.990	14.812	14.075	14.275	14.659
-5.802	0.085	1.601	0.289	-0.003	0.001	6.510	5.999	6.738	6.224	14.787	14.209	14.246	14.594
-5.786	0.098	1.590	-0.260	-0.026	-0.021	6.362	6.625	6.635	5.835	14.863	13.948	14.285	14.763
-5.796	0.088	1.598	0.006	0.273	-0.004	6.500	6.322	6.618	6.171	14.752	13.858	14.331	14.335
-5.800	0.080	1.602	-0.000	-0.201	-0.004	6.345	6.327	6.753	5.892	14.827	14.248	14.267	14.871
-0.030	4.056	1.566	0.005	0.001	-0.008	12.276	12.052	7.907	8.206	8.643	7.579	12.284	12.181
-0.036	4.035	1.575	0.273	0.007	0.000	12.339	11.770	7.937	8.228	8.591	7.687	12.236	11.994
-0.038	4.041	1.571	-0.071	0.040	0.004	12.268	12.112	7.887	8.240	8.664	7.524	12.294	12.286
-0.032	4.048	1.568	0.003	0.287	-0.002	12.314	12.216	7.869	8.528	8.555	7.400	12.342	11.945
-0.018	0.032	1.181	-0.004	-0.001	-0.005	9.976	9.348	9.651	9.661	10.004	9.357	9.727	9.732
0.005	4.023	1.567	0.025	-0.280	-0.027	12.212	11.948	7.983	7.880	8.689	7.834	12.257	12.407
0.004	4.012	1.564	0.008	-0.001	0.087	12.345	11.991	7.943	8.163	8.692	7.540	12.338	12.122
0.005	3.997	1.569	-0.004	0.004	-0.796	11.810	12.481	7.809	8.364	8.179	7.983	11.926	12.554
5.970	-0.003	1.575	-0.002	0.002	0.006	14.960	14.224	14.442	14.826	6.408	6.302	6.678	5.928
5.960	-0.020	1.578	0.348	0.018	-0.016	15.030	14.048	14.452	14.971	6.338	6.718	6.585	5.727
5.925	0.004	1.577	-0.010	-0.330	-0.016	14.833	13.949	14.548	14.417	6.500	6.317	6.592	6.172
5.972	-0.003	1.568	-0.007	-0.003	-0.495	14.629	14.509	14.289	15.022	6.203	6.506	6.455	6.199
0.004	-3.972	1.589	-0.001	-0.001	-0.000	8.603	7.526	12.283	12.171	12.312	12.069	7.923	8.207
0.026	-3.901	1.593	-0.039	-0.019	-0.015	8.607	7.566	12.226	12.107	12.370	11.978	7.910	8.221
0.001	-3.963	1.588	-0.000	-0.290	-0.012	8.504	7.353	12.342	11.944	12.340	12.235	7.885	8.543
0.007	-3.964	1.589	-0.000	-0.013	-0.394	8.385	7.714	12.081	12.415	12.063	12.289	7.830	8.332

TAB. D.4 – Jeu de poses d'étalonnage du robot CoGiRo  $S_3$  (en mètres et en radians)

$x$	$y$	$z$	$\psi$	$\theta$	$\phi$	$l_1$	$l_2$	$l_3$	$l_4$	$l_5$	$l_6$	$l_7$	$l_8$
-5.965	-3.884	1.259	0.008	-0.016	0.092	4.586	3.606	10.355	9.581	16.840	16.192	13.553	14.018
-3.286	-3.955	2.495	-0.350	-0.013	-0.011	5.216	4.507	10.454	9.927	14.457	13.758	10.613	11.195
3.174	-3.975	1.240	0.081	0.002	-0.002	11.664	10.570	14.389	14.511	10.752	10.808	5.688	5.571
-2.791	-1.791	2.486	0.280	-0.008	-0.004	6.461	5.559	8.838	8.835	12.719	12.455	10.622	11.030
-5.974	-3.983	2.502	-0.190	-0.011	-0.006	3.355	2.746	9.933	9.211	16.548	15.970	13.202	13.824
3.006	-2.978	2.465	-0.378	-0.015	-0.006	11.166	10.443	13.242	13.297	9.676	9.177	5.413	5.555

TAB. D.5 – Jeu de poses d'étalonnage du robot CoGiRo  $S_4$  (en mètres et en radians)

$x$	$y$	$z$	$\psi$	$\theta$	$\phi$	$l_1$	$l_2$	$l_3$	$l_4$	$l_5$	$l_6$	$l_7$	$l_8$
-0.029	0.076	1.179	-0.005	-0.000	0.000	9.975	9.347	9.651	9.662	10.005	9.356	9.727	9.731
-4.022	3.070	1.227	0.008	-0.003	-0.011	9.580	9.538	5.571	5.228	12.554	11.591	14.364	14.520
-4.023	3.070	1.647	0.004	-0.002	-0.012	9.399	9.403	5.256	4.976	12.415	11.478	14.245	14.434
2.966	0.064	1.640	-0.010	0.005	-0.804	11.737	12.051	11.559	12.343	7.251	7.653	7.339	7.759
2.965	0.064	1.225	-0.010	0.006	-0.800	11.881	12.155	11.706	12.448	7.481	7.817	7.569	7.919
-0.030	0.071	1.182	-0.004	-0.000	-0.002	9.971	9.346	9.648	9.664	10.003	9.357	9.724	9.731
2.963	0.067	1.221	-0.011	0.006	-0.799	11.883	12.156	11.706	12.447	7.487	7.821	7.569	7.919
2.966	0.071	1.634	-0.010	0.005	-0.804	11.740	12.052	11.561	12.346	7.252	7.653	7.341	7.761
-4.020	3.069	1.649	0.005	-0.005	-0.014	9.395	9.401	5.255	4.972	12.409	11.477	14.241	14.431
-4.020	3.071	1.228	0.008	-0.005	-0.011	9.578	9.537	5.570	5.223	12.549	11.589	14.366	14.519
-0.026	0.076	1.180	-0.005	-0.004	-0.001	9.974	9.347	9.650	9.659	10.003	9.356	9.723	9.734
-4.019	3.070	1.661	0.009	-0.003	-0.013	9.395	9.398	5.253	4.974	12.407	11.474	14.242	14.430
2.965	0.069	1.654	-0.010	0.004	-0.801	11.736	12.048	11.557	12.341	7.248	7.650	7.337	7.757

TAB. D.6 – Jeu de poses d'étalonnage du robot CoGiRo  $S_5$  (en mètres et en radians)

# Index

Étalonnage, [20](#)

Amélioration de la précision, [18](#)

Analyse par intervalles, [28](#)

Câbles, [62](#)

Certification, [109](#)

Contracteurs, [35](#)

Géométrico-statique, [65](#)

Identifiabilité, [84](#), [106](#)

Implicite, [23](#)

Incertitudes, [19](#)

Manipulateurs, [5](#)

Manipulateurs parallèles à câbles, [8](#)

    Pleinement contraint, [11](#)

    Suspendu, [11](#)

Mesures aberrantes, [45](#)

Moindres carrés, [24](#)

Poses optimales, [86](#)

Préconditionnement, [33](#)



---

## Certified calibration of parallel cable-driven robots

**Abstract :** The main objective of this thesis is to propose new methods for the calibration of a large scale cable-driven robot.

The principal method to improve the global behavior of a robot consists to identify the parameters of the model. For this, it is important to get redundant information by measuring the state of the robot in different configurations. However, the model used is a compromise between its ability to represent the actual behavior of the manipulator and the information available to fill in it.

In the special case of the large scale cable-driven robots, mass and elasticity of the cables have a significant influence on the behavior of the robot but they are difficult to model. Indeed, the physical model of the cable is complex and requires knowledge of the tension inside it. Available sensors can not provide this information with a sufficient accuracy to fill in a model of a realistic cable, we thus propose to use a simplified model. In order to provide an efficient calibration, it is necessary to fix the requirements to use this simplified model. Then, we have adapted and implemented some classical techniques for the calibration of parallel robots, but we also developed more innovative approaches. We propose a model for cable robots based on a representation of the uncertainties from modeling, measurements and parameters using intervals. By exploiting the interval analysis, we have developed various approaches to identify with certification the geometric parameters of the structure. We thus propose a new approach and associated algorithms to characterize and compute different kind of solutions for the calibration problem. In addition, based on recent results obtained with constraint programming, this thesis describes a new method for the identification of parameters, robust to outliers and modeling uncertainties.

The methods defined in this thesis were implemented and applied to the calibration of the large scale parallel cable-driven robot built in the CoGiRo national project.

**Keywords :** Parallel robots, wire, cable, calibration, interval analysis

---



---

## Étalonnage certifié des robots parallèles à câbles

**Résumé :** L'objectif de cette thèse est de proposer des méthodes pour étalonner un robot parallèle à câbles de grande dimension.

Afin d'améliorer le comportement global d'un robot, il est nécessaire d'identifier au mieux les paramètres de son modèle. Pour cela, il est important d'obtenir des informations redondantes en mesurant l'état du robot dans différentes configurations. Cependant, le modèle choisi est un compromis entre sa capacité à représenter le comportement réel du manipulateur et les informations disponibles pour le renseigner.

Dans le cas particulier des robots à câbles de grande dimension, la masse et l'élasticité des câbles ont une influence non négligeable sur le comportement du robot mais sont difficiles à modéliser. En effet, le modèle physique des câbles est complexe et nécessite de connaître la tension à laquelle ils sont soumis. Les capteurs disponibles ne pouvant nous fournir cette information avec une précision suffisante pour renseigner un modèle de câble réaliste, nous proposons d'utiliser un modèle simplifié. Dans le but de proposer un étalonnage efficace, il est donc nécessaire de définir les conditions pour l'emploi de ce modèle simplifié. Ensuite, nous avons adapté et implanté d'une part plusieurs techniques classiques pour l'étalonnage des robots parallèles mais nous avons également élaboré des approches plus innovantes. Nous proposons en effet un modèle pour les robots à câbles reposant sur une représentation des incertitudes de modélisation, de mesures et de paramètres au moyen d'intervalles. En exploitant l'analyse par intervalles, nous avons pu élaborer différentes approches pour identifier les paramètres géométriques de la structure et cela de manière certifiée. Nous proposons ainsi une nouvelle approche et les algorithmes correspondants afin de caractériser et de calculer différents types de solutions à l'étalonnage. De plus, en se fondant sur certains résultats récents obtenus avec la programmation par contraintes, cette thèse décrit une nouvelle méthode pour l'identification de paramètres, robuste aux mesures aberrantes et aux incertitudes de modélisation.

Les méthodes définies dans cette thèse ont été implantées et appliquées à l'étalonnage du robot parallèle à câbles de grande dimension construit dans le cadre de l'ANR CoGiRo.

**Mots clés :** Robots parallèles, câbles, étalonnage, analyse par intervalles

---